I declare under penalty of perjury under the laws of the United States that the foregoing is true and correct.

Executed this 16th day of November, 2007 at Menlo Park, California.

Wei Wei

EXHIBIT A

Exponent
Failure Analysis Associates®

Exponent
149 Commonwealth Drive
Menlo Park, CA 94025

telephone 650-326-9400
facsimile 650-326-8072
www.exponent.com

Wei Wei
Managing Engineer

Professional Profile

Mr. Wei Wei is a Managing Engineer in Exponent's Electrical and Semiconductors practice. He has extensive experience in the area of silicon integrated circuit (IC) design, verification, and fabrication. His experience covers logic and embedded memory IC design, physical layout, and IC photo-mask production. He is also knowledgeable in areas related to electronics systems design and manufacturing, semiconductor manufacturing processes and device physics.

Prior to joining Exponent, Mr. Wei was a project leader at a major semiconductor company, where he had led the research and development work of several flash memory based IC devices such as Programmable Logic Devices (PLDs) and configuration memories for Field Programmable Gate Arrays (FPGAs). He had successfully produced several commercial IC products and also worked extensively on related customer or yield issues, such as Electrostatic Discharge (ESD) failures, noise problems, flash memory data retentions, etc.

At Exponent, Mr. Wei specializes in investigations of semiconductor component failures and reliability problems, such as ESD/latch up suceptability, Time Dependant Dielectric Breakdown (TDDB) lifetime, package moisture sensitivity, etc. In addition, he had conducted extensive investigations of failures related to other electronics components and Printed Circuit Boards (PCBs), such as capacitor failures, electrochemical migration, PCB propagating faults, etc. Mr. Wei had also conducted a number of product safety reviews on electronics products, performing tasks such as component stress level analysis, circuit board failure mode and effect analysis, and customized laboratory testing. He had also been involved in several intellectual property litigation cases related to semiconductor IC design and manufacturing.

Credentials and Professional Honors

M.S., Electrical Engineering, University of Washington, 1999
B.S., Microelectronics, Peking University, Beijing, China, 1997

Member: IEEE (SSC, EDS, CPMT societies), ESDA

Patents

"Sense Amplifier with Reduced Low Power Mode Sense Delay," application filed 2004.

Presentations

"A PWB Failure Case Study," CTIA Battery Registration Program Ad-Hoc Meeting, Menlo Park, California, USA, January, 2007

EXHIBIT B

16 Mbit LPC Serial Flash
SST49LF016C

Data Sheet

FEATURES:

- **Operational Clock Frequency**
  - 33 MHz
  - 66 MHz
- **Organized as 2M x8**
- **Conforms to LPC Interface Specification v1.1**
  - Support Multi-Byte Firmware Memory Read/ Write Cycles
- **Single 3.0-3.6V Read and Write Operations**
- **LPC Mode**
  - 5-signal LPC bus interface for both in-system and factory programming using programmer equipment
  - Multi-Byte Read data transfer rate 15.6 MB/s @ 33 MHz PCI clock and 31.2 MB/s @ 66 MHz clock
    - Firmware Memory Read cycle supporting 1, 2, 4, 16, and 128 Byte Read
    - Firmware Memory Write cycle supporting 1, 2, and 4 Byte Write
  - 33 MHz/66 MHz clock frequency operation
  - WP#/AAI and TBL# pins provide hardware Write protect for entire chip and/or top Boot Block
  - Block Locking Registers for individual block Read-Lock, Write-Lock, and Lock-Down protection
  - 5 GPI pins for system design flexibility
  - 4 ID pins for multi-chip selection
  - Multi-Byte capability registers (read-only registers)
  - Status register for End-of-Write detection
  - Program-/Erase-Suspend Read or Write to other blocks during Program-/Erase-Suspend
- **Two-cycle Command Set**
- **Security ID Feature**
  - 256-bit Secure ID space
    - 64-bit Unique Factory Pre-programmed Device Identifier
    - 192-bit User-Programmable OTP
- **Superior Reliability**
  - Endurance: 100,000 Cycles (typical)
  - Greater than 100 years Data Retention
- **Low Power Consumption**
  - Active Read Current: 12 mA (typical)
  - Standby Current: 10 µA (typical)
- **Uniform 4 KByte sectors**
  - 35 Overlay Blocks: one 16-KByte Boot Block, two 8-KByte Parameter Blocks, one 32-Kbyte Parameter Block, thirty-one 64-KByte Main Blocks.
- **Fast Sector-Erase/Program Operation**
  - Sector-Erase Time: 18 ms (typical)
  - Block-Erase Time: 18 ms (typical)
  - Program Time: 7 µs (typical)
- **Auto Address Increment (AAI) for Rapid Factory Programming (High Voltage Enabled)**
  - RY/BY# pin for End-of-Write detection
  - Multi-Byte Program
  - Chip Rewrite Time: 4 seconds (typical)
- **Packages Available**
  - 32-lead PLCC
  - 32-lead TSOP (8mm x 14mm)
- **All non-Pb (lead-free) devices are RoHS compliant**

PRODUCT DESCRIPTION

The SST49LF016C flash memory device is designed to interface with host controllers (chipsets) that support a low-pin-count (LPC) interface for system firmware applications. Complying with LPC Interface Specification 1.1, SST49LF016C supports a Burst-Read data transfer of 15.6 MBytes per second at 33 MHz clock speed and 31.2 MBytes per second at 66 MHz clock speed, up to 128 bytes in a single operation.

The LPC interface operates with 5 signal pins versus 28 pins of a 8-bit parallel flash memory. This frees up pins on the ASIC host controller resulting in lower ASIC costs and a reduction in overall system costs due to simplified signal routing. This 5-signal LPC interface supports both in-system and rapid factory programming using programmer equipment. A high voltage pin (WP#/AAI) enables Auto Address Increment (AAI) mode.

Via the software registers, the SST49LF016C offers hardware block protection and individual block protection for critical system code and data. The 256-bit Security ID space is comprised of a 64-bit factory pre-programmed unique number and a 192-bit One-Time-Programmable (OTP) area. This Security ID permits the use of new security techniques and implementation of a new data protection scheme. To protect against inadvertent write, the SST49LF016C device has on-chip hardware and software write protection schemes. The SST49LF016C also provides general purpose inputs (GPI) for system design flexibility.

Manufactured with SST proprietary, high-performance SuperFlash technology, SST49LF016C has a split-gate cell design and thick-oxide tunneling injector for greater reliability and manufacturability compared with alternative technology approaches.

The SST49LF016C significantly improves performance and reliability, while lowering power consumption. The total energy consumed is a function of the applied voltage, current and time of application. Because the SST49LF016C writes in-system with a single 3.0-3.6V power supply, the total energy consumed during any Erase or Program operation is less than alternative flash memory technologies.

The SuperFlash technology provides fixed Erase and Program time, independent of the number of Erase/Program cycles performed. This feature eliminates system software or hardware calibration or erase cycle correlation which is necessary with alternative flash memory technologies, whose Erase and Program time increase with accumulated Erase/Program cycles.

The SST49LF016C product provides a maximum program time of 10 µs per byte with a single-byte Program operation; effectively 5 µs per byte with a dual-byte Program operation and 2.5 µs per byte with a quad-byte Program operation.

The SST49LF016C is offered in 32-PLCC and 32-TSOP packages. See Figures 3 and 4 for pin assignments and Table 1 for pin descriptions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

FUNCTIONAL BLOCKS

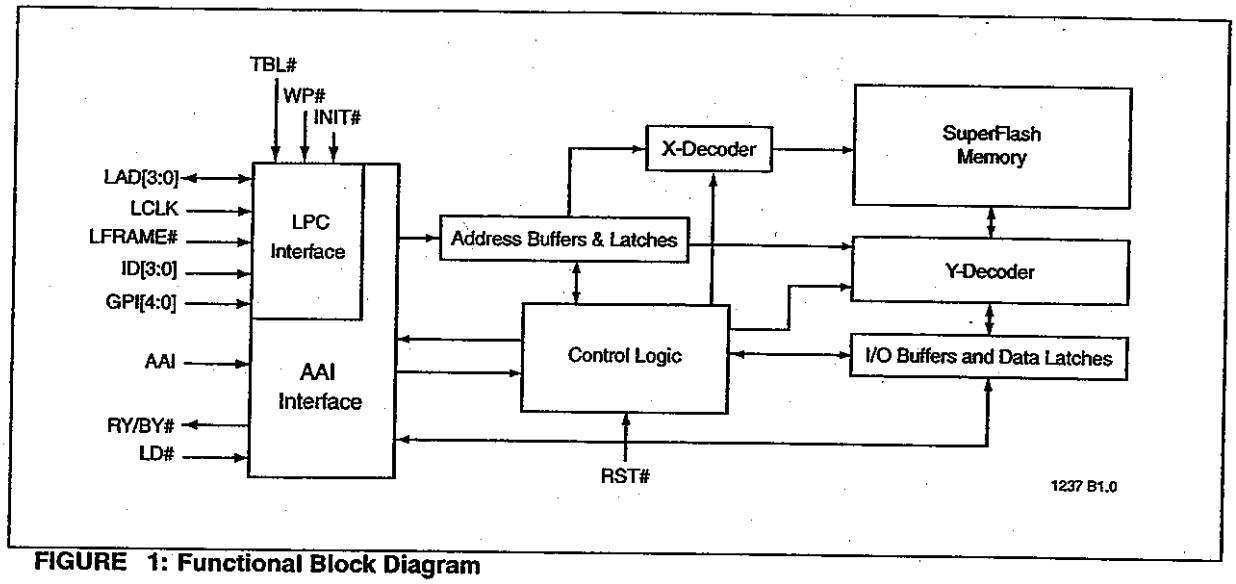


FIGURE 1: Functional Block Diagram

DEVICE MEMORY MAP

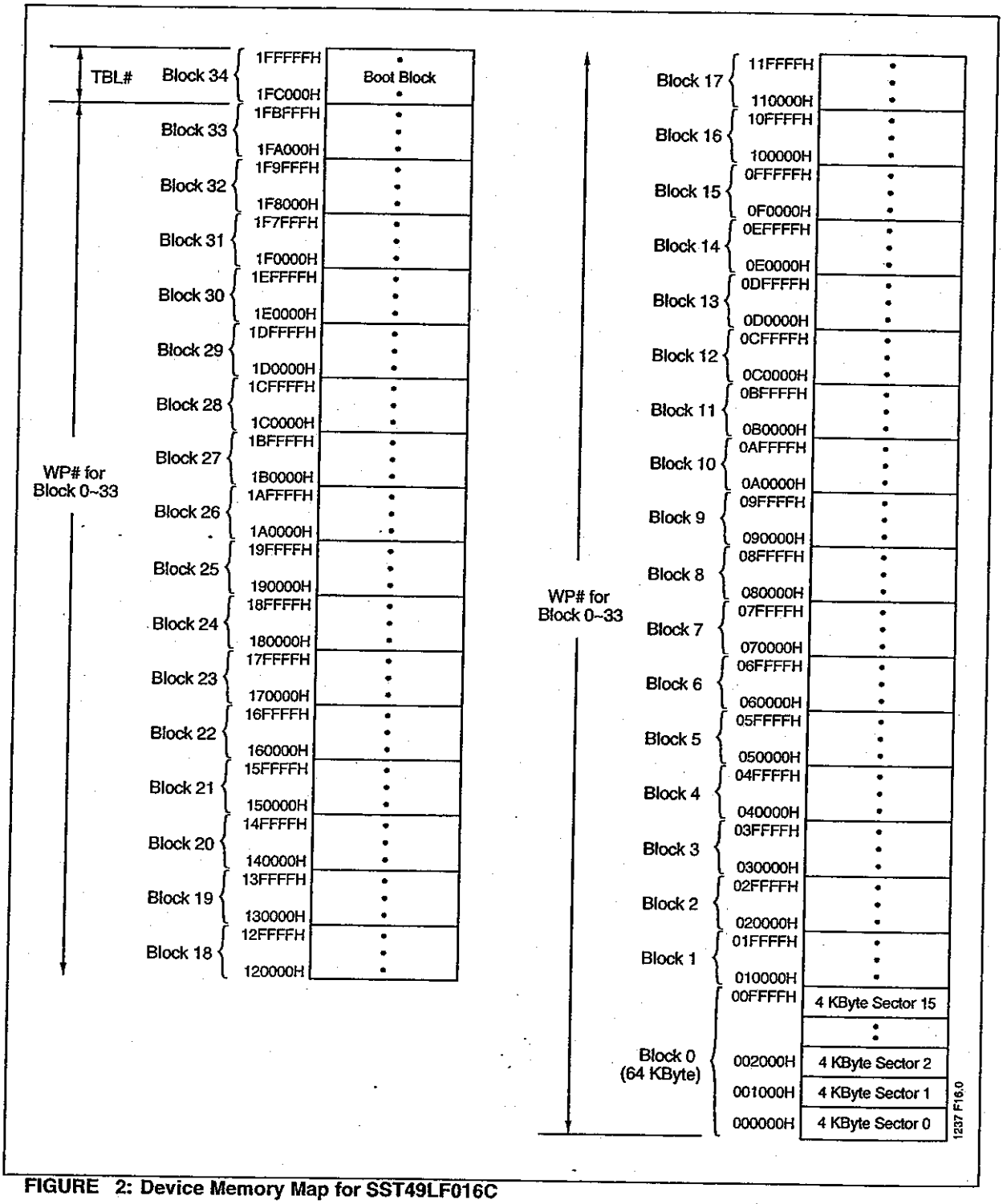


FIGURE 2: Device Memory Map for SST49LF016C

PIN ASSIGNMENTS

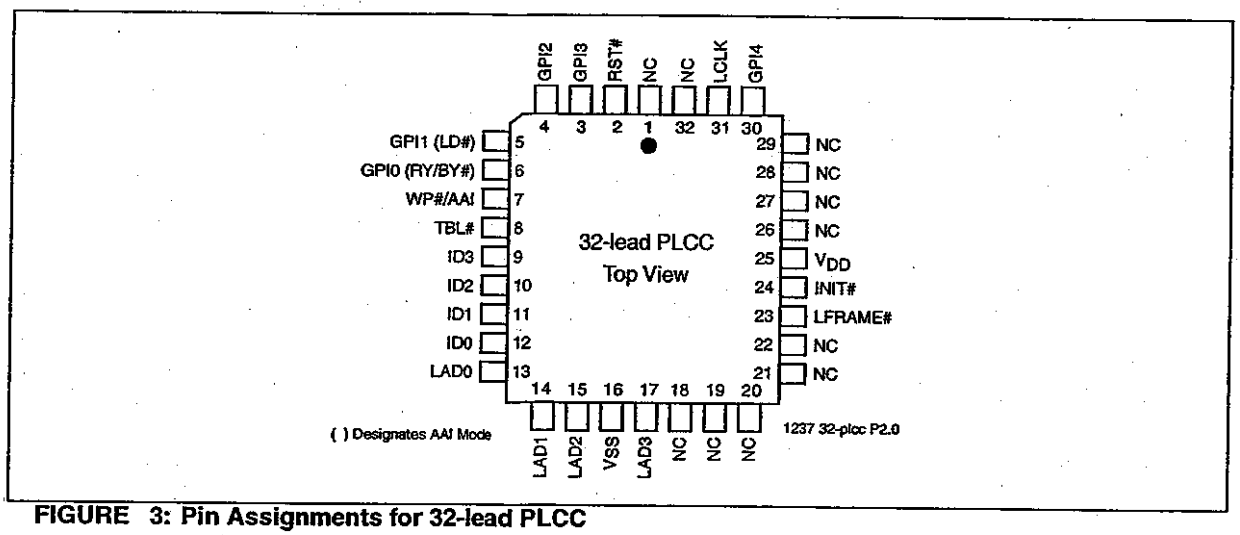


FIGURE 3: Pin Assignments for 32-lead PLCC

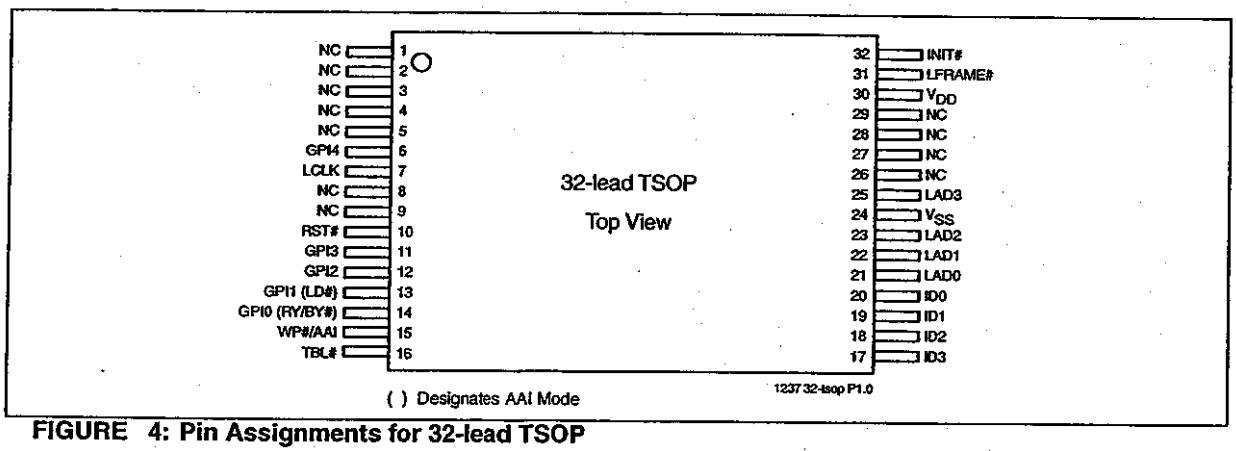


FIGURE 4: Pin Assignments for 32-lead TSOP

PIN DESCRIPTIONS

TABLE 1: Pin Description

Symbol	Pin Name	Type[1]	Interface		Functions
			AAI	LPC	
LCLK	Clock	I	X	X	To accept a clock input from the control unit
LAD[3:0]	Address and Data	I/O	X	X	To provide LPC bus information, such as addresses and command Inputs/ Outputs data.
LFRAME#	Frame	I	X	X	To indicate the start of a data transfer operation; also used to abort an LPC cycle in progress.
RST#	Reset	I	X	X	To reset the operation of the device
INIT#	Initialize	I	X	X	This is the second reset pin for in-system use. This pin is internally combined with the RST# pin. If this pin or RST# pin is driven low, identical operation is exhibited.
ID[3:0]	Identification Inputs	I	X	X	These four pins are part of the mechanism that allows multiple parts to be attached to the same bus. The strapping of these pins is used to identify the component. The boot device must have ID[3:0]=0000, all subsequent devices should use sequential up-count strapping. These pins are internally pulled-down with a resistor between 20-100 KΩ. When in AAI mode, these pins operate identically as in Firmware Memory cycles.
GPI[4:0]	General Purpose Inputs	I		X	These individual inputs can be used for additional board flexibility. The state of these pins can be read through LPC registers. These inputs should be at their desired state before the start of the LPC clock cycle during which the read is attempted, and should remain in place until the end of the Read cycle. Unused GPI pins must not be floated. GPI[2:4] are ignored when in AAI mode.
TBL#	Top Block Lock	I		X	When low, prevents programming to the boot block sectors at top of device memory. When TBL# is high it disables hardware write protection for the top block sectors. This pin cannot be left unconnected. TBL# setting is ignored when in AAI mode.
WP#/AAI	Write Protect	I		X	When low, prevents programming to all but the highest addressable block (Boot Block). When WP# is high it disables hardware write protection for these blocks. This pin cannot be left unconnected.
WP#/AAI	AAI Enable	I	X		When set to the Supervoltage V_H = 9V, configures the device to program multiple bytes in AAI mode. When brought to V_{IL}/V_{IH}, returns device to LPC mode.
RY/BY#	Ready/Busy#	O	X		Open drain output that indicates the device is ready to accept data in an AAI mode, or that the internal cycle is complete. Used in conjunction with LD# pin to switch between these two flag states.
LD#	Load-Enable#	I	X		Input pin which when low, indicates the host is loading data in an AAI programming cycle. If LD# is high, the host signals the AAI interface that it is terminating a command. LD# low/high switches the RY/BY# output from a "buffer free" flag to a "programming complete" flag.
V_{DD}	Power Supply	PWR	X	X	To provide power supply (3.0-3.6V)
V_{SS}	Ground	PWR	X	X	Circuit ground (0V reference)
NC	No Connection		N/A	N/A	Unconnected pins.

1. I=Input, O=Output

T1.2 1237

Clock

The LCLK pin accepts a clock input from the host controller.

Input/Output Communications

The LAD[3:0] pins are used to serially communicate cycle information such as cycle type, cycle direction, ID selection, address, data, and sync fields.

Input Communication Frame

The LFRAME# pin is used to indicate start of a LPC bus cycle. The pin is also used to abort an LPC bus cycle in progress.

Reset

A V_{IL} on INIT# or RST# pin initiates a device reset. INIT# and RST# pins have the same function internally. It is required to drive INIT# or RST# pins low during a system reset to ensure proper CPU initialization. During a Read operation, driving INIT# or RST# pins low deselects the device and places the output drivers, LAD[3:0], in a high impedance state. The reset signal must be held low for a minimum of time T_{RSTP}. A reset latency occurs if a reset procedure is performed during a Program or Erase operation. See Table 25, Reset Timing Parameters, for more information. A device reset during an active Program or Erase operation will abort the operation and memory contents may become invalid due to data being altered or corrupted from an incomplete Erase or Program operation.

Identification Inputs

These pins are part of a mechanism that allows multiple devices to be attached to the same bus. The strapping of these pins is used to identify the component. The boot device must have ID[3:0] = 0; all subsequent devices should use sequential count-up strapping. These pins are internally pulled-down with a resistor between 20-100 KΩ.

General Purpose Inputs

The General Purpose Inputs (GPI[4:0]) can be used as digital inputs for the CPU to read. The GPI register holds the values on these pins. The data on the GPI pins must be stable before the start of a GPI register Read and remain stable until the Read cycle is complete. The pins must be driven low, V_{IL}, or high, V_{IH} but not left unconnected (float).

Write Protect / Top Block Lock

The Top Boot Lock (TBL#) and Write Protect (WP#/AAI) pins are provided for hardware write protection of device memory in the SST49LF016C. The TBL# pin is used to write protect 16 KByte at the highest memory address range for the SST49LF016C. WP#/AAI pin write protects the remaining sectors in the flash memory. An active low signal at the TBL# pin prevents Program and Erase operations of the top Boot Block. When TBL# pin is held high, write protection of the top Boot Block is then determined by the Boot Block Locking registers. The WP#/AAI pin serves the same function for the remaining sectors of the device memory. The TBL# and WP#/AAI pins write protection functions operate independently of one another. Both TBL# and WP#/AAI pins must be set to their required protection states prior to starting a Program or Erase operation. A logic level change occurring at the TBL# or WP#/AAI pin during a Program or Erase operation could cause unpredictable results. TBL# and WP#/AAI pins cannot be left unconnected.

TBL# is internally OR'ed with the top Boot Block Locking register. When TBL# is low, the top Boot Block is hardware write protected regardless of the state of the Write-Lock bit for the Boot Block Locking register. Clearing the Write-Protect bit in the register when TBL# is low will have no functional effect, even though the register may indicate that the block is no longer locked.

WP#/AAI is internally OR'ed with the Block Locking register. When WP#/AAI is low, the blocks are hardware write protected regardless of the state of the Write-Lock bit for the corresponding Block Locking registers. Clearing the Write-Protect bit in any register when WP#/AAI is low will have no functional effect, even though the register may indicate that the block is no longer locked.

AAI Enable

The AAI Enable pin (WP#/AAI) is used to enable the Auto Address Increment (AAI) mode. When the WP#/AAI pin is set to the Supervoltage V_H (9±0.5V), the device is in AAI mode with Multi-Byte programming. When the WP#/AAI pin is brought to V_{IL}/V_{IH} levels, the device returns to LPC mode.

Ready/Busy

The Ready/Busy pin (RY/BY#), is an open drain output which indicates the device is ready to accept data in an AAI mode, or that the internal programming cycle is complete. The pin is used in conjunction with the LD# pin to switch between these two flag states (see Table 18).

Load Enable

The Load Enable pin (LD#), is an input pin which when low, indicates the host is loading data in an AAI programming cycle. Data is loaded in the SST49LF016C at the rising edge of the clock. If LD# is high, it signals the AAI interface that the host is terminating the command. LD# low/high switches the RY/BY# output from buffer free flag to programming complete flag (see Table 18).

No Connection (NC)

These pins are not connected internally.

DESIGN CONSIDERATIONS

SST recommends a high frequency 0.1 µF ceramic capacitor to be placed as close as possible between V_{DD} and V_{SS} less than 1 cm away from the V_{DD} pin of the device. Additionally, a low frequency 4.7 µF electrolytic capacitor from V_{DD} to V_{SS} should be placed within 1 cm of the V_{DD} pin. If you use a socket for programming purposes add an additional 1-10 µF next to each socket. The RST# pin must remain stable at V_{IH} for the entire duration of an Erase operation. WP#/AAI must remain stable at V_{IH} for the entire duration of the Erase and Program operations for non-Boot Block sectors. To write data to the top Boot Block sectors, the TBL# pin must also remain stable at V_{IH} for the entire duration of the Erase and Program operations.

MODE SELECTION

The SST49LF016C flash memory device operates in two distinct interface modes: the LPC mode and the Auto Address Increment (AAI) mode. The WP#/AAI pin is used to set the interface mode selection. The device is in AAI mode when the WP#/AAI pin is set to the Supervoltage V_H (9±0.5V), and in the LPC mode when the WP#/AAI is set to V_{IL}/V_{IH}. The mode selection must be configured prior to device operation.

LPC MODE

Device Operation

The SST49LF016C supports Multi-Byte Firmware Memory Read and Write cycle types as defined in Low Pin Count Interface Specification, Revision 1.1. Table 2 shows the size of transfer supported by the SST49LF016C.

TABLE 2: Transfer Size Supported

Cycle Type	Size of Transfer
Firmware Memory Read	1, 2, 4, 16, 128 Bytes
Firmware Memory Write	1, 2, 4 Bytes

T2.1 1237

The LPC mode uses a 5-signal communication interface: one control line, LFRAME#, which is driven by the host to start or abort a bus cycle, a 4-bit data bus, LAD[3:0], used to communicate cycle type, cycle direction, ID selection, address, data and sync fields. The device enters standby mode when LFRAME# is taken high and no internal operation is in progress.

The host drives LFRAME# signal from low-to-high to capture the start field of a LPC cycle. On the cycle in which LFRAME# goes inactive, the last latched value is taken as the START value. The START value determines whether the SST49LF016C will respond to a Firmware Memory Read/Write cycle type as defined in Table 3.

TABLE 3: Firmware Memory Cycles START Field Definition

START Value	Definition
1101	Start of a Firmware Memory Read cycle
1110	Start of a Firmware Memory Write cycle

T3.1 1237

See following sections on details of Firmware Memory cycle types (Tables 4 and 5). Two-cycle Program and Erase command sequences are used to initiate Firmware Memory Program and Erase operations. See Table 8 for a listing of Program and Erase commands.

FIRMWARE MEMORY CYCLES

Firmware Memory Read Cycle

TABLE 4: Firmware Memory Read Cycle Field Definitions

Clock Cycle	Field Name	Field Contents LAD[3:0][1]	LAD[3:0] Direction	Comments
1	START	1101	IN	LFRAME# must be active (low) for the part to respond. Only the last start field (before LFRAME# transitions high) will be recognized. The START field contents (1101b) indicate a Firmware Memory Read cycle.
2	IDSEL	0000 to 1111	IN	Indicates which SST49LF016C device should respond. If the IDSEL (ID select) field matches the value of ID[3:0], then that particular device will respond to the LPC bus cycle.
3-9	MADDR	YYYY	IN	These seven clock cycles make up the 28-bit memory address. YYYY is one nibble of the entire address. Addresses are transferred most-significant nibble first.
10	MSIZE	KKKK	IN	The MSIZE field indicates how many bytes will be transferred during multi-byte operations. Device will execute multi-byte read of 2^{MSIZE} bytes. SST49LF016C supports only MSIZE = 0, 1, 2, 4, 7 (1, 2, 4, 16, 128 Bytes), with KKKK=0000b, 0001b, 0010b, 0100b, or 0111b.
11	TAR0	1111	IN, then Float	In this clock cycle, the master has driven the bus to all '1's and then floats the bus, prior to the next clock cycle. This is the first part of the bus "turnaround cycle."
12	TAR1	1111 (float)	Float, then OUT	The SST49LF016C takes control of the bus during this cycle.
13	RSYNC	0000 (READY)	OUT	During this clock cycle, the device generates a "ready sync" (RSYNC) indicating that the device has received the input data. The least-significant nibble of the least-significant byte will be available during the next clock cycle.
14-A	DATA	ZZZZ	OUT	A=(13+2^{n+1}); n = MSIZE Least significant nibbles outputs first.
(A+1)	TAR0	1111	OUT, then Float	In this clock cycle, the SST49LF016C drives the bus to all ones and then floats the bus prior to the next clock cycle. This is the first part of the bus "turnaround cycle." A=(13+2^{n+1}); n = MSIZE
(A+2)	TAR1	1111 (float)	Float, then IN	The host resumes control of the bus during this cycle. A=(13+2^{n+1}); n = MSIZE

T4.0 1237

1. Field contents are valid on the rising edge of the present clock cycle.

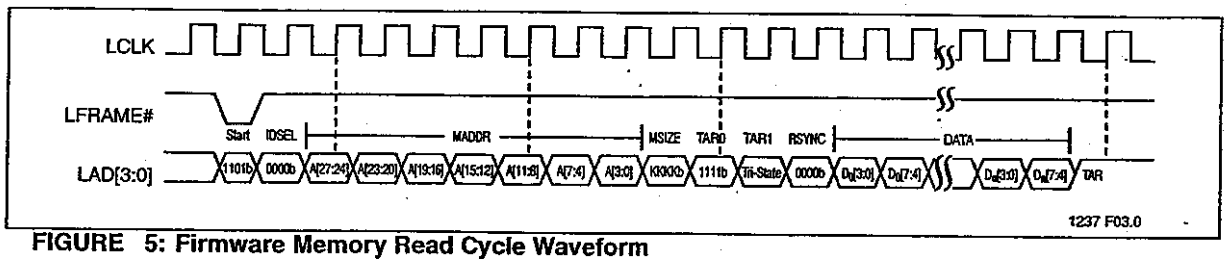


FIGURE 5: Firmware Memory Read Cycle Waveform

Firmware Memory Write Cycle

TABLE 5: Firmware Memory Write Cycle

Clock Cycle	Field Name	Field Contents LAD[3:0][1]	LAD[3:0] Direction	Comments
1	START	1110	IN	LFRAME# must be active (low) for the part to respond. Only the last start field (before LFRAME# transitions high) will be recognized. The START field contents (1110b) indicate a Firmware Memory Write cycle.
2	IDSEL	0000 to 1111	IN	Indicates which SST49LF016C device should respond. If the IDSEL (ID select) field matches the value of ID[3:0], then that particular device will respond to the whole bus cycle.
3-9	MADDR	YYYY	IN	These seven clock cycles make up the 28-bit memory address. YYYY is one nibble of the entire address. Addresses are transferred most-significant nibble first.
10	MSIZE	KKKK	IN	The MSIZE field indicates how many bytes will be transferred during multi-byte operations. Device supports 1, 2, and 4 Bytes write with MSIZE = 0, 1, or 2, and KKKK=0000b, 0001b, or 0010b.
11-A	DATA	ZZZZ	IN	A=(10+2^{n+1}); n = MSIZE Least significant nibble entered first.
(A+1)	TAR0	1111	IN then Float	In this clock cycle, the master has driven the bus to all '1's and then floats the bus prior to the next clock cycle. This is the first part of the bus "turnaround cycle." A=(10+2^{n+1}); n = MSIZE
(A+2)	TAR1	1111 (float)	Float then OUT	The SST49LF016C takes control of the bus during this cycle. A=(10+2^{n+1}); n = MSIZE
(A+3)	RSYNC	0000	OUT	During this clock cycle, the SST49LF016C generates a "ready sync" (RSYNC) and outputs the values 0000, indicating that it has received data or a flash command. A=(10+2^{n+1}); n = MSIZE
(A+4)	TAR0	1111	OUT then Float	In this clock cycle, the SST49LF016C drives the bus to all '1's and then floats the bus prior to the next clock cycle. This is the first part of the bus "turnaround cycle". A=(10+2^{n+1}); n = MSIZE
(A+5)	TAR1	1111 (float)	Float then IN	The host resumes control of the bus during this cycle. A=(10+2^{n+1}); n = MSIZE

T5.0 1237

1. Field contents are valid on the rising edge of the present clock cycle.

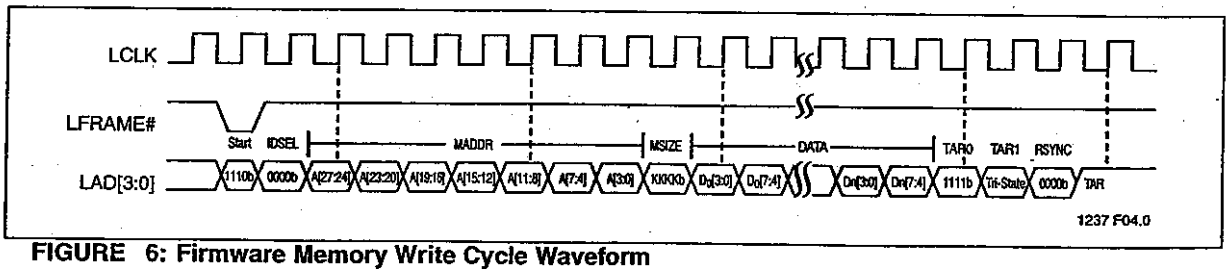


FIGURE 6: Firmware Memory Write Cycle Waveform

Abort Mechanism

If LFRAME# is driven low for one or more clock cycles after the start of a bus cycle, the cycle will be terminated. The host may drive the LAD[3:0] with '1111b' (ABORT nibble) to return the interface to ready mode. The ABORT only affects the current bus cycle. For a multi-cycle command sequence, such as the Erase or Program commands, ABORT doesn't interrupt the entire command sequence, only the current bus cycle of the command sequence. The host can re-send the bus cycle for the aborted command and continue the command sequence after the device is ready again.

Response to Invalid Fields for Firmware Memory Cycle

During an on-going Firmware Memory bus cycle, the SST49LF016C will not explicitly indicate that it has received invalid field sequences. The response to specific invalid fields or sequences is described as follows:

ID mismatch: If the IDSEL field does not match ID[3:0], the device will ignore the cycle. See "Multiple Device Selection for Firmware Memory Cycle" on page 15 for details.

Address out of range: The address sequence is 7 fields long (28 bits) with Firmware Memory bus cycles. Only some of the address fields bits are decoded by the SST49LF016C. These are: A_0 through A_{20} and A_{22}. Address A_{22} has the special function of directing reads and writes to the flash core (A_{22}=1) or to the register space (A_{22}=0).

Invalid MSIZE field: If the SST49LF016C receives an invalid size field during a Firmware Memory Read or Write operation, the device will reset and no operation will be attempted. The device will not generate any kind of response in this situation. The SST49LF016C will only respond to values listed in Table 6.

TABLE 6: Valid MSIZE field Values for Firmware Memory Cycles

MSIZE	Direction	Size of Transfer
0000	R/W	1 Byte
0001	R/W	2 Byte
0010	R/W	4 Byte
0100	R	16 Byte
0111	R	128 Byte

T6.0 1237

Once valid START, IDSEL, and MSIZE are received, the SST49LF016C will always complete the bus cycle. However, if the device is busy performing a flash Erase or Program operation, no new internal memory Write will be executed. As long as the states of LAD[3:0] and LFRAME# are known, the response of the ST49LF016C to signals received during the cycle is predictable.

Non-boundary-aligned address: The SST49LF016C accepts multi-byte transfers for both Read and Write operations. The device address space is divided into uniform page sizes 2, 4, 16, or 128 bytes wide, according to the MSIZE value (see Table 6). The host issues only one address in the MADDR field of the Firmware Memory Cycle, but multiple bytes are read from or written to the device. For this reason the MADDR address should be page boundary-aligned. This means the address should be aligned to a Word boundary (A_0 = 0) for a 2-byte transfer, a double Word boundary (e.g. A_0 = 0, A_1 = 0) for a 4-byte transfer, and so on. If the address supplied by the host is not page boundary-aligned, the SST49LF016C will force a boundary alignment, starting the multi-byte Read or Write operation from the lower byte of the addressed page.

Multiple Device Selection

Multiple LPC serial flash devices may be strapped to increase memory densities in a system. The four ID pins, ID[3:0], allow up to 16 devices to be attached to the same bus by using different ID strapping in a system. BIOS support, bus loading, or the attaching bridge may limit this number. The boot device must have an ID of 0000b (determined by ID[3:0]); subsequent devices use incremental numbering. Equal density must be used with multiple devices.

Multiple Device Selection for Firmware Memory Cycle

For Firmware Memory Read/Write cycles, hardware strapping values on ID[3:0] must match the values in IDSEL field. The SST49LF016C will compare these bits with ID[3:0]'s strapping values. If there is a mismatch, the device will ignore the remainder of the cycle. See Table 7 for Multiple Device Selection Configuration.

TABLE 7: Firmware Memory Multiple Device Selection Configuration

Device #	ID[3:0]	IDSEL
0 (Boot device)	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	1001
10	1010	1010
11	1011	1011
12	1100	1100
13	1101	1101
14	1110	1110
15	1111	1111

T7.0 1237

DEVICE COMMANDS

Device operation is controlled by commands written to the Command User Interface (CUI). Execution of a specific command is handled by internal functions after a CUI receives and processes the command. After power-up or a Reset operation the device enters Read mode. Commands consist of one or two sequential Bus-Write operations. The commands are summarized in Table 8, "Software Command Sequence".

TABLE 8: Software Command Sequence

Command	Bus Cycles Required	First Bus Cycle			Second Bus Cycle		
		Oper	Addr[1]	Data	Oper	Addr[1]	Data
Read-Array/Reset	1	Write	X	FFH			
Read-Software-ID[2]/ Read-Security-ID[3]	≥ 2	Write	X	90H	Read	IA[4]	ID[5]
Read-Status-Register[3]	2	Write	X	70H	Read	X	SRD[6]
Clear-Status-Register	1	Write	X	50H			
Sector-Erase[7]	2	Write	X	30H	Write	SAx[8]	D0H
Block-Erase[7]	2	Write	X	20H	Write	BAx	D0H
Program[7,9]	2	Write	X	40H or 10H	Write	WA[10]	WD[11]
Program-/Erase-Suspend	1	Write	X	B0H			
Program-/Erase-Resume	1	Write	X	D0H			
User-Security-ID-Program[12]	2	Write	X	A5H	Write	WA[10]	Data
User-Security-ID-Program-Lockout	2	Write	X	85H	Write	X	00H

T8.0 1237

1. This value must be a valid address within the device Memory Address Space. X can be V_{IH} or V_{IL}, but no other value.
2. SST Manufacturer's ID = BFH, is read with A_{20}-A_0 = 0.
   SST49LF016C Device ID = 5CH, is read with A_{20}-A_1 = 0, A_0 = 1.
   Following the Read-Software-ID/Read-Security-ID command,
   Read operations access Manufacturer's ID and Device ID or Security ID.
3. Following the Read-Software-ID/Read-Security-ID command, Read operations access manufacturer's ID and Device ID or Security ID. Read-Software-ID/Read-Security-ID and Read-Status-Register will return register data until another valid command is written.
4. IA = Device Identification Address/Security ID Address.
5. ID = Data read from identifier codes/Data read from Security ID
6. SRD = Data read from Status register
7. The sector or block must not be write-locked when attempting Erase or Program operations.
   Attempts to issue an Erase or Program command to a write-locked sector/block will fail.
8. SA_X for Sector-Erase Address
   BA_X for Block-Erase Address
9. The Program command operates on multiple bytes.
10. WA = Address of memory location to be written
11. WD = Data to be written at location WA
12. Valid addresses for the User Security ID space are from FFFC 0188H to FFFC 019FH.

Read-Array Command

Upon initial device power-up and after exit from reset, the device defaults to the read array mode. This operation can also be initiated by writing the Read-Array command. (See Table 8.) The device remains available for array reads until another command is written. Once an internal Program/Erase operation starts, the device will not recognize the Read-Array command until the operation is completed, unless the operation is suspended via a Program/Erase Suspend command.

Read-Software-ID Command

The Read-Software-ID operation is initiated by writing the Read-Software-ID command. Following the command, the device will output the manufacturer's ID and device ID from the addresses shown in Table 9. Any other valid command will terminate the Read-Software-ID operation.

The Read-Software-ID command is the same as the Read-Security-ID command. See "Security ID Commands" on page 19.

TABLE 9: Product Identification

	Address[1]	Data
Manufacturer's ID	FFFC 0000H	BFH
Device ID SST49LF016C	FFFC 0001H	5CH

T9.1 1237

1. Address shown in this column is for boot device only. Address locations should appear elsewhere in the 4 GByte system memory map depending on ID strapping values on ID[3:0] pins when multiple LPC memory devices are used in a system.

Read-Status-Register Command

The Status register may be read to determine when a Sector-/Block-Erase or Program completes, and whether the operation completed successfully. The Status register may be read at any time by writing the Read-Status-Register command. After writing this command, all subsequent Read operations will return data from the Status register until another valid command is written.

The default value of the Status register after device power-up or reset is 80H.

Clear-Status-Register Command

The user can reset the Status register's Block Protect Status (BPS) bit to 0 by issuing a Clear-Status-Register command. Device power-up and hardware reset will also reset BPS to 0.

TABLE 10: Software Status Register

Bit	Name	Function
0	RES	Reserved for future use
1	BPS	Block Protect Status The Block Write-Lock bit should be interrogated only after Erase or Program command is issued. It informs the system whether or not the selected block is locked. BPS does not provide a continuous indication of Write-Lock bit value. 0: Block Unlocked 1: Operation Aborted, Block Write-Lock bit set.
2:5	RES	Reserved for future use
6	ESS	Erase Suspend Status 0: Erase in progress/completed 1: Erase suspended
7	WSMS	Write State Machine Status Check WSMS to determine erase or program completion. 0: Busy 1: Ready

T10.0 1237

Sector-/Block-Erase Command

The Erase Command operates on one sector or block at a time. This command requires an (arbitrary) address within the sector or block to be erased. Note that a Sector/Block Erase operation changes all Sector/Block byte data to FFh. If a Read operation is performed after issuing the erase command, the device will automatically output Status Register data. The system can poll the Status Register in order to verify the completion of the Sector/Block Erase operation (please refer to Table 10, Status Register Definition). If a Sector/Block Erase is attempted on a locked block, the operation will fail and the data in the Sector/Block will not be changed. In this case, the Status Register will report the error (BPS=1).

Program Command

The Program command operates on multiple bytes (Refer to Table 5). This command specifies the address and data to be programmed. During the Program operation the device automatically outputs the Status Register data when read. The system can poll the Status Register in order to verify the completion of the Program operation (refer to Table 10, "Software Status Register"). If a Program operation is attempted on a locked block, the operation will fail and the data in the addressed byte will not be changed. In this case, the Status Register will report the error (BPS=1).

Program-/Erase-Suspend or Program-/Erase-Resume Operations

The Program-Suspend and Erase-Suspend operations share the same software command sequence (B0H). The Program-Resume and Erase-Resume operations share the same software command sequence (D0H). See Table 8, "Software Command Sequence" on page 16.

Erase-Suspend/ Erase-Resume Commands

The Erase Suspend command allows Sector-Erase or Block-Erase interruption in order to read or program data in another block of memory. Once the Erase-Suspend command is executed, the device will suspend any ongoing Erase operation within time T_{ES} (10 µs). The device outputs status register data when read after the Erase-Suspend command is written. The system is able to determine when the Erase operation has been completed (WSMS=1) by polling the status register. After an Erase-Suspend, the device will set the status register ESS bit (ESS=1) if the Erase has been successfully suspended (refer to Table 10, "Software Status Register"). The Erase-Resume command resumes the Erase operation that had been previously suspended.

After a successful Erase-Suspend, a Read-Array command may be written to read data from a Sector/Block other than the suspended Sector/Block. A Program command sequence may also be issued during Erase Suspend to program data in memory locations other than the Sector/Block currently in the Erase-Suspend mode. If a Read-Array command is written to an address within the suspended Sector/Block this may result in reading invalid data. If a Program command is written to an address within the suspended Sector/Block the command is acknowledged but rejected. Other valid commands while erase is suspended include Read-Status-Register, Read-Device-ID, and Erase-Resume.

The Erase-Resume command resumes the Erase process in the suspended sector or block. After the Erase-Resume command is written, the device will continue the Erase process. Erase cannot resume until any Program operation initiated during Erase-Suspend has completed. Suspended operations cannot be nested: the system needs to complete or resume any previously suspended operation before a new operation can be suspended. See Figure 7 for flowchart.

Program-Suspend/ Program-Resume Command

The Program-Suspend and Program-Resume commands have no influence on the device. Since the device requires a maximum of T_{BP} (10 µs) in order to program a byte (see Table 26), when a Program-Suspend command is written, the suspended Byte Program operation will always be successfully completed within the suspend latency time ($T_{ES} = T_{BP} = 10$ µs).

Security ID Commands

The SST49LF016C device offers a 256-bit Security ID space. The Security ID space is divided into two parts. One 64-bit segment is programmed at SST with a unique 64-bit number: this number cannot be changed by the user. The other segment is 192-bit wide and is left blank: this space is available for customers and can be programmed as desired.

The User-Security-ID-Program command is shown in Table 8, "Software Command Sequence". Use the memory addresses specified in Table 11 for Security ID programming. Once the customer segment is programmed, it can be locked to prevent any alteration. The User-Security-ID-Program-Lockout command is shown in Table 8, "Software Command Sequence".

In order to read the Security ID information, the user can issue a Read Security ID Command (90H) to the device. At this point the device enters the Read-Software-ID/Read-Security-ID mode. The Security ID information can be read at the memory addresses in Table 11.

A Read-Array/Reset command (FFH) must then be issued to the device in order to exit the Read-Software-ID/Read-Security-ID mode and return to Read-Array mode.

An alternate method to read the Security ID information is to read the Security ID registers located into the register space as described in the "Security ID Registers" section.

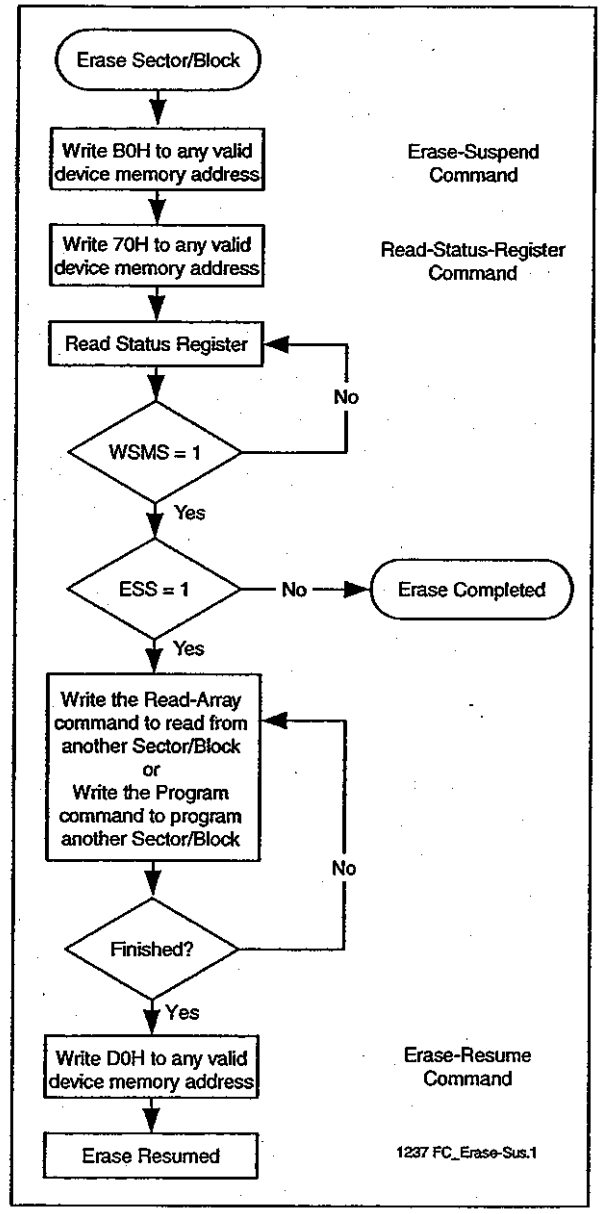


FIGURE 7: Erase-Suspend Flow Chart

TABLE 11: Security ID Addresses

Address Range	Security ID Segment	Size
FFFC 0180 to FFFC 0187	Factory-Programmed	8 bytes – 64 bit
FFFC 0188 to FFFC 019F	User-Programmed	24 bytes – 192 bit

T11.0 1237

REGISTERS

There are five types of registers available on the SST49LF016C, the multi-byte Read/Write configuration registers (for Firmware Memory cycle), General Purpose Inputs registers, Block Locking registers, Security ID register, and the JEDEC ID registers. These registers appear at their respective address location in the 4 GByte system memory map. Unused register locations will read as 00H. Any attempt to read or write any register during an internal Write operation will be ignored.

Read or write access to the register during an internal Program/Erase operation will be completed as follows:

- Multi-byte Read/Write Configuration registers, General Purpose Inputs register, and Block Locking registers can be accessed normally
- Security ID register and the JEDEC ID registers can not be accessed (reading these registers will return unused register data 00H).

Multi-Byte Read/Write Configuration Registers (Firmware Memory Cycle)

The multi-byte read/write configuration (MBR) registers are four 8-bit read-only registers located at addresses FFBC0005-FFBC0008 for boot configured device (see Table 13). These registers are accessible using Firmware Memory Read cycle only. These registers contain information about multi-byte read and write access sizes that will be accepted for Firmware Memory multi-byte Read commands. The registers are not available in AAI mode.

In case of multi-byte Firmware Memory register reads, the device will return register data for the addressed register until the command finishes, or is aborted.

General Purpose Inputs Register

The General Purpose Inputs register (GPI_REG) passes the state of GPI[4:0] pins on the SST49LF016C. It is recommended that the GPI[4:0] pins be in the desired state before LFRAME# is brought low for the beginning of the bus cycle, and remain in that state until the end of the cycle. There is no default value since this is a pass-through register. The GPI_REG register for the boot device appears at FFBC0100H in the 4 GByte system memory map, and will appear elsewhere if the device is not the boot device (see Table 12). This register is not available to be read when the device is in an Erase/Program operation. In case of multi-byte Firmware Memory cycle register reads, the device will return register data for the addressed register until the command finishes, or is aborted.

TABLE 12: General Purpose Register

Register	Register Address[1]	Default Value	Access
GPI_REG	FFBC 0100H	N/A	R

T12.0 1237

1. Address shown in this column is for boot device only. Address locations should appear elsewhere in the 4 GByte system memory map depending on ID strapping values on ID[3:0] pins when multiple LPC memory devices are used in a system.

TABLE 13: Multi-byte Read/Write Configuration Registers (Firmware Memory Cycle Only)

Register	Register Address[1]	Data	Access	Description
MULTI_BYTE_READ_L	FFBC 0005H	0100 1011b	R	Device supports 1,2,4, 16, 128 Byte reads
MULTI_BYTE_READ_H	FFBC 0006H	0000 0000b	R	Future Expansion for Read
MULTI_BYTE_WRITE_L	FFBC 0007H	0000 0011b	R	Device supports 1, 2, 4 Byte Write
MULTI_BYTE_WRITE_H	FFBC 0008H	0000 0000b	R	Future Expansion for Write

T13.0 1237

1. Address shown in this column is for boot device only. Address locations should appear elsewhere in the 4 GByte system memory map depending on ID strapping values on ID[3:0] pins when multiple LPC memory devices are used in a system.

Block Locking Registers

SST49LF016C provides software controlled lock protection through a set of Block Locking registers. The Block Locking Registers are read/write registers and they are accessible through standard addressable memory locations specified in Table 14. Unused register locations will return 00H if read.

In case of multi-byte register reads with Firmware Memory cycle, the device will return register data for the addressed register until the command finishes, or is aborted.

TABLE 14: Block Locking Registers

Register	Block Size	SST49LF016C Protected Memory Address[1] Range	Memory Map Register Address[1]
T_BLOCK_LK	16K	1FFFFFH-1FC000H	FFBFC002H
T_MINUS01_LK	8K	1FBFFFH-1FA000H	FFBFA002H
T_MINUS02_LK	8K	1F9FFFH-1F8000H	FFBF8002H
T_MINUS03_LK	32K	1F7FFFH-1F0000H	FFBF0002H
T_MINUS04_LK	64K	1EFFFFH-1E0000H	FFBE0002H
T_MINUS05_LK	64K	1DFFFFH-1D0000H	FFBD0002H
T_MINUS06_LK	64K	1CFFFFH-1C0000H	FFBC0002H
T_MINUS07_LK	64K	1BFFFFH-1B0000H	FFBB0002H
T_MINUS08_LK	64K	1AFFFFH-1A0000H	FFBA0002H
T_MINUS09_LK	64K	19FFFFH-190000H	FFB90002H
T_MINUS10_LK	64K	18FFFFH-180000H	FFB80002H
T_MINUS11_LK	64K	17FFFFH-170000H	FFB70002H
T_MINUS12_LK	64K	16FFFFH-160000H	FFB60002H
T_MINUS13_LK	64K	15FFFFH-150000H	FFB50002H
T_MINUS14_LK	64K	14FFFFH-140000H	FFB40002H
T_MINUS15_LK	64K	13FFFFH-130000H	FFB30002H
T_MINUS16_LK	64K	12FFFFH-120000H	FFB20002H
T_MINUS17_LK	64K	11FFFFH-110000H	FFB10002H
T_MINUS18_LK	64K	10FFFFH-100000H	FFB00002H
T_MINUS19_LK	64K	0FFFFFH-0F0000H	FFAF0002H
T_MINUS20_LK	64K	0EFFFFH-0E0000H	FFAE0002H
T_MINUS21_LK	64K	0DFFFFH-0D0000H	FFAD0002H
T_MINUS22_LK	64K	0CFFFFH-0C0000H	FFAC0002H
T_MINUS23_LK	64K	0BFFFFH-0B0000H	FFAB0002H
T_MINUS24_LK	64K	0AFFFFH-0A0000H	FFAA0002H
T_MINUS25_LK	64K	09FFFFH-090000H	FFA90002H
T_MINUS26_LK	64K	08FFFFH-080000H	FFA80002H
T_MINUS27_LK	64K	07FFFFH-070000H	FFA70002H
T_MINUS28_LK	64K	06FFFFH-060000H	FFA60002H
T_MINUS29_LK	64K	05FFFFH-050000H	FFA50002H
T_MINUS30_LK	64K	04FFFFH-040000H	FFA40002H
T_MINUS31_LK	64K	03FFFFH-030000H	FFA30002H
T_MINUS32_LK	64K	02FFFFH-020000H	FFA20002H
T_MINUS33_LK	64K	01FFFFH-010000H	FFA10002H
T_MINUS34_LK	64K	00FFFFH-000000H	FFA00002H

T14.0 1237

1. Address shown in this column is for boot device only. Address locations should appear elsewhere in the 4 GByte system memory map depending on ID strapping values on ID[3:0] pins when multiple LPC memory devices are used in a system.

TABLE 15: Block Locking Register Bits

Reserved Bit [7:3]	Read-Lock Bit [2]	Lock-Down Bit [1]	Write-Lock Bit [0]	Lock Status
00000	0	0	0	Full Access
00000	0	0	1	Write Locked (Default State at Power-Up)
00000	0	1	0	Locked Open (Full Access Locked Down)
00000	0	1	1	Write Locked Down
00000	1	0	0	Block Read Locked (Registers alterable)
00000	1	0	1	Block Read & Write Lock (Registers alterable)
00000	1	1	0	Block Read Locked Down (Registers not alterable)
00000	1	1	1	Block Read & Write lock Down (Registers not alterable)

T15.0 1237

Write-Lock Bit

The Write-Lock bit, bit 0, controls the lock state described in Table 15. The default Write status of all blocks after power up is write locked. When bit 0 of the Block Locking register is set, Program and Erase operations for the corresponding block are prevented. Clearing the Write-Lock bit will unprotect the block. The Write-Lock bit must be cleared prior to starting a Program or Erase operation since it is sampled at the beginning of the operation. The Write-Lock bit functions in conjunction with the hardware Write Lock pin TBL# for the top Boot Block. When TBL# is low, it overrides the software locking scheme. The top Boot Block Locking register does not indicate the state of the TBL# pin. The Write-Lock bit functions in conjunction with the hardware WP#/AAI pin for the remaining blocks (Blocks 0 to 33 for 49LF016C). When WP#/AAI is low, it overrides the software locking scheme. The Block Locking register does not indicate the state of the WP#/AAI pin.

Lock-Down Bit

The Lock-Down bit, bit 1, controls the Block Locking register as described in Table 15. When in LPC interface mode, the default Lock Down status of all blocks upon power-up is not locked down. Once the Lock-Down bit is set, any future attempted changes to that Block Locking register will be ignored. The Lock-Down bit is only cleared upon a device reset with RST# or INIT# or power down. Current Lock Down status of a particular block can be determined by reading the corresponding Lock-Down bit. Once a block's Lock-Down bit is set, the Read-Lock and Write-Lock bits for that block can no longer be modified: the block is locked down in its current state of read/write accessibility.

Read-Lock Bit

The default read status of all blocks upon power-up is read-unlocked. When a block's read lock bit is set, data cannot be read from that block. An attempted read from a read-locked block will result in the data 00h. The read lock status can be unlocked by clearing the read lock bit: this can only be done provided that the block is not locked down. The current read lock status of a particular block can be determined by reading the corresponding read-lock bit.

Security ID Registers

The SST49LF016C device offers a 256-bit Security ID register space. The Security ID space is divided into two segments - one (64-bits) factory programmed segment and one (192 bits) user programmed segment. The first segment is programmed and locked at SST with a unique 64-bit number. The user segment (192 bits) is left blank (FFH) for the customer to be programmed as desired. Refer to Table 8, "Software Command Sequence" for more details.

The Security ID Information and its Write Lock/Unlock status can be Read in the Register Access Space for Execute-In-Place type of applications. (See Table 16.)

The Write Lock-out status of the Security ID space can be read from the SEC_ID_WRITE_LOCK register (see Table 16). The SEC_ID_WRITE_LOCK register is a read-only register that is accessible at the address location specified in Table 16.

In case of multi-byte register reads with Firmware Memory cycle, for SEC_ID_WRITE_LOCK register, the device will return register data for the addressed register until the command finishes, or is aborted.

In the case of multi-byte register reads with Firmware Memory cycle, for all the SEC_ID_BYTE registers, the device will return page-aligned sequential register data with wrap-around until the command finishes, or is aborted.

TABLE 16: Security ID Registers

Register	Register Address[1]	Value	Access	Description
SEC_ID__WRITE_LOCK	FFBC0102H	0000 0000b 0000 0001b	R	Write Unlocked Write Locked
SEC_ID_BYTE_0	FFBC0180H		R	Factory Programmed
SEC_ID_BYTE_1	FFBC0181H		R	Factory Programmed
SEC_ID_BYTE_2	FFBC0182H		R	Factory Programmed
SEC_ID_BYTE_3	FFBC0183H		R	Factory Programmed
...	...		...	...
SEC_ID_BYTE_7	FFBC0187H		R	Factory Programmed
SEC_ID_BYTE_8	FFBC0188H		R	User Programmed
SEC_ID_BYTE_9	FFBC0189H		R	User Programmed
...	...		...	...
SEC_ID_BYTE_30	FFBC019EH		R	User Programmed
SEC_ID_BYTE_31	FFBC019FH		R	User Programmed

T16.0 1237

1. Address shown in this column is for boot device only. Address locations should appear elsewhere in the 4 GByte system memory map depending on ID strapping values on ID[3:0] pins when multiple LPC memory devices are used in a system.

JEDEC ID Registers

The JEDEC ID registers for the boot device appear at FFBC0000H and FFBC0001H in the 4 GByte system memory map, and will appear elsewhere if the device is not the boot device. This register is not available to be read when the device is in Erase/Program operation. Unused register location will read as 00H. See Table 17 for the JEDEC device ID code. In case of multi-byte register reads with Firmware Memory cycle, the device will return register data for the addressed register until the command finishes, or is aborted.

TABLE 17: JEDEC ID Registers

Register	Register Address[1]	Default Value	Access
MANUF_REG	FFBC 0000H	BFH	R
DEV_REG	FFBC 0001H	5CH	R

T17.0 1237

1. Address shown in this column is for boot device only. Address locations should appear elsewhere in the 4 GByte system memory map depending on ID strapping values on ID[3:0] pins when multiple LPC memory devices are used in a system.

AUTO-ADDRESS INCREMENT (AAI) MODE

AAI Mode with Multi-byte Programming

AAI mode with multi-byte programming is provided for high-speed production programming. Auto-Address Increment mode requires only one address load for each 128-byte page of data.

Taking the WP#/AAI pin to the Supervoltage V_H enables the AAI mode. The AAI command is started as a normal Firmware Memory cycle. LD# should be low (V_{IL}) as long as data is being loaded into the device. In the MADDR field, the host may input any address within the 128-byte page to be programmed. The least significant seven bits of the address field will be ignored and the device will begin programming at the beginning of the 128-byte page (i.e., the address will be page-aligned). The device Ready/Busy status is output on the RY/BY# pin.

Data is accepted until the internal buffer is full. At that point RY/BY# goes low (busy) to indicate that the internal buffer is full and cannot accept any more data. When the device is ready, RY/BY# pin goes high and indicates to the host that more data (the next group of bytes) can be accepted by the internal data buffer (see Table 18 and Figure 8).

After loading the final byte(s) of the 128-byte page, the RY/BY# signal remains low until the completion of internal programming. After the completion of programming, the part will go into idle mode and the RY/BY# will go high indicating that the AAI command has been completed (see Table 18). A subsequent AAI command may be initiated to begin programming the next 128-byte page.

Data will be accepted by the device as long as LD# is low and RY/BY# is high (until the last byte of the 128-byte page has been entered). For partial data-loads (i.e., less than 128 Bytes), LD# may be taken high (V_{IH}) to end the data loading. If LD# goes high before the full 128-byte page has been entered, the device will program the data which has been entered to that point, and then terminate the AAI page programming command. Any incompletely loaded data byte (nibble) will not be programmed. The device will signify completion of the command by driving RYBY# high. Once RY/BY# goes high, LD# can be taken low to begin a new AAI programming operation at a different address location.

The RY/BY# pin will stay low while internal programming completes. When the entire 128-byte page has been programmed, the device will return to the idle mode and the RY/BY# pin will go high (V_{IH}) to indicate the AAI command has been completed.

TABLE 18: LD# Input and RY/BY# Status in AAI Mode

LD# state	RY/BY# status	RY/BY# Flag indication
L	H	Device is Ready, can accept more data until the last (128th) byte.
L	L	Device is Busy, cannot accept more data
L	H	Device is Ready for next operation if previous data is the last (128th) byte.
H	H	Device is Ready for next operation
H	L	Device is Busy programming

T18.1 1237

The user may terminate AAI programming by dropping the WP#/AAI pin to TTL levels (V_{IH}/V_{IL}) as long as LD# is high and RY/BY# returns to high indicating the completion of the AAI cycle. Software block-locking will be disabled in AAI mode (all blocks will be write-unlocked). If AAI drops below the Supervoltage V_H before RY/BY# returns to high (and LD# high), the contents of the page may be indeterminate.

AAI Data Load Protocol

TABLE 19: AAI Programming Cycle (initiated with WP#/AAI at V_H ONLY)

Clock Cycle	Field Name	Field Contents	LAD[3:0]	Comments
1	START	1110	IN	LFRAME# must be active (low) for the part to respond. Only the last start field (before LFRAME# transitions high) should be recognized. The START field contents indicate a Firmware Memory Write cycle. (1110b)
2	IDSEL	0000b to 1111b	IN	ID works identically to Firmware Memory cycle. This field indicates which SST49LF016C device should respond. If the IDSEL (ID select) field matches the value of ID[3:0], then that particular device will respond to the whole bus cycle.
3-9	MADDR	YYYY	IN	These seven clock cycles make up the 28-bit memory address. YYYY is one nibble of the entire address. Addresses are transferred most-significant nibble first. Only bits [20:7] of the total address [27:0] are used for AAI mode. The rest are "don't care".
10	MSIZE	KKKK	IN	MSIZE field is don't care when in AAI mode
11-266	DATA	ZZZZ	IN	Data is transmitted to the device least significant nibble first, from byte 0 to byte 127 as long as the RY/BY# is high and LD# low. The host will pause the clock and data stream when RY/BY# goes low until it returns high, signifying that the chip is ready for more data

T19.0 1237

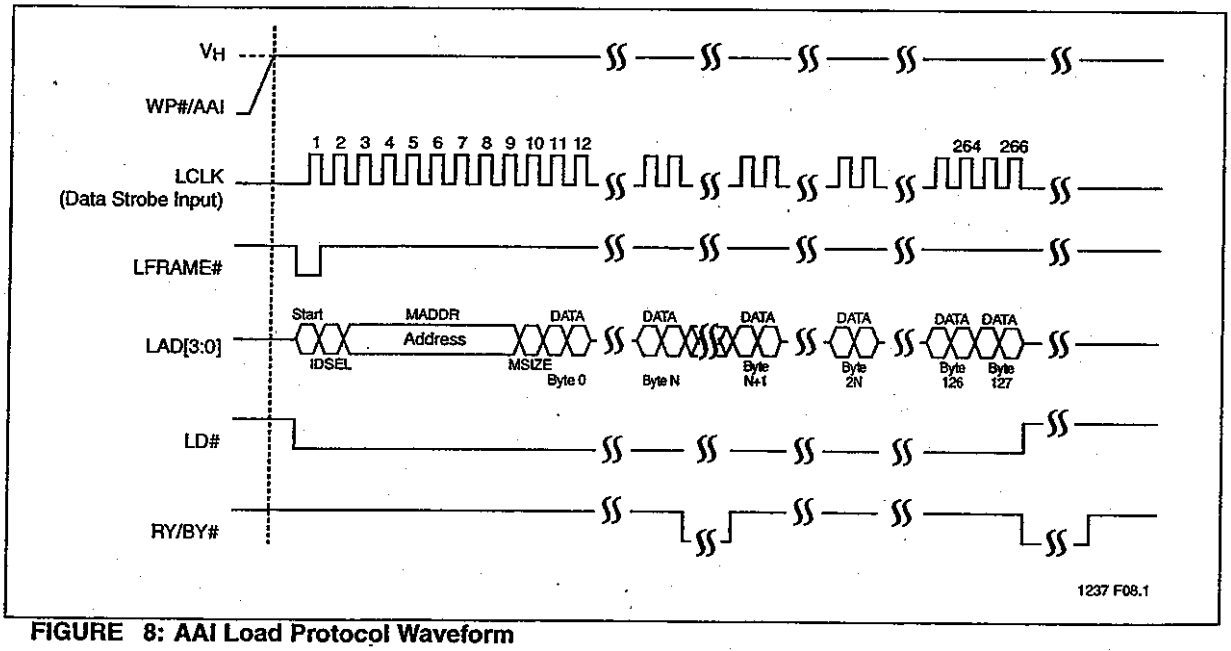


FIGURE 8: AAI Load Protocol Waveform

ELECTRICAL SPECIFICATIONS

The AC and DC specifications for the LPC interface signals (LAD[3:0], LFRAME#, LCLCK and RST#) as defined in Section 4.2.2.4 of the PCI local Bus specification, Rev. 2.1. Refer to Table 20 for the DC voltage and current specifications. Refer to Table 24 through Table 26 for the AC timing specifications for Clock, Read, Write, and Reset operations.

Absolute Maximum Stress Ratings (Applied conditions greater than those listed under "Absolute Maximum Stress Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these conditions or conditions greater than those defined in the operational sections of this data sheet is not implied. Exposure to absolute maximum stress rating conditions may affect device reliability.)

Temperature Under Bias . . . . . -55°C to +125°C
Storage Temperature . . . . . -65°C to +150°C
D.C. Voltage on Any Pin to Ground Potential . . . . . -0.5V to V_{DD}+0.5V
Transient Voltage (<20 ns) on Any Pin (except WP#/AAI) to Ground Potential[1] . . . . . -2.0V to V_{DD}+2.0V
Voltage on WP#/AAI Pin to Ground Potential[2] . . . . . -0.5V to 11.0V
Package Power Dissipation Capability (T_A=25°C) 1.0W
Surface Mount Solder Reflow Temperature[3] . . . . . 260°C for 10 seconds
Output Short Circuit Current[4] . . . . . 50 mA

1. Do not violate processor or chipset specification regarding INIT# voltage.
2. The maximum DC voltage on WP#/AAI pin may reach 11V for periods <20ns.
3. Excluding certain with-Pb 32-PLCC units, all packages are 260°C capable in both non-Pb and with-Pb solder versions. Certain with-Pb 32-PLCC package types are capable of 240°C for 10 seconds; please consult the factory for the latest information.
4. Outputs shorted for no more than one second. No more than one output shorted at a time.

OPERATING RANGE

Range	Ambient Temp	V_{DD}
Commercial	0°C to +85°C	3.0-3.6V

AC CONDITIONS OF TEST

Input Rise/Fall Time	3 ns
Output Load	C_L = 30 pF
See Figures 14 and 15	

DC Characteristics

TABLE 20: DC Operating Characteristics at 33 MHz and 66 MHz (All Interfaces)

Symbol	Parameter	Limits			Test Conditions
		Min	Max	Units	
I_{DD}[1]	Active V_{DD} Current				LCLK (LPC mode)=V_{ILT}/V_{IHT} All other inputs=V_{IL} or V_{IH}
	Read		18	mA	All outputs = open, V_{DD}=V_{DD} Max
	Single-/Dual-Byte Program, Erase		40	mA	
	Quad-Byte Program		60	mA	
I_{SB}	Standby V_{DD} Current (LPC Interface)		100	µA	LCLK (LPC mode)=V_{ILT}/V_{IHT} at LFRAME#=.9V_{DD} V_{DD}=V_{DD} Max All other inputs ≥ 0.9 V_{DD} or ≤ 0.1 V_{DD}
I_{RY}[2]	Ready Mode V_{DD} Current		10	mA	LCLK (LPC mode)=V_{ILT}/V_{IHT} LFRAME#=V_{IL}, V_{DD}=V_{DD} Max All other inputs ≥ 0.9 V_{DD} or ≤ 0.1 V_{DD}
I_I	Input Leakage Current for ID[3:0] pins		200	µA	V_{IN}=GND to V_{DD}, V_{DD}=V_{DD} Max
I_{LI}	Input Leakage Current		1	µA	V_{IN}=GND to V_{DD}, V_{DD}=V_{DD} Max
I_{LO}	Output Leakage Current		1	µA	V_{OUT}=GND to V_{DD}, V_{DD}=V_{DD} Max
I_H	Supervoltage Current for WP#/AAI		200	µA	
V_H	Supervoltage for WP#/AAI	8.5	9.5	V	
V_{IHI}[3]	INIT# Input High Voltage	1.1	V_{DD}+0.5	V	V_{DD}=V_{DD} Max
V_{ILI}[3]	INIT# Input Low Voltage	-0.5	0.4	V	V_{DD}=V_{DD} Min
V_{IL}	Input Low Voltage	-0.5	0.3 V_{DD}	V	V_{DD}=V_{DD} Min
V_{IH}	Input High Voltage	0.5 V_{DD}	V_{DD}+0.5	V	V_{DD}=V_{DD} Max
V_{OL}	Output Low Voltage		0.1 V_{DD}	V	I_{OL}=1500 µA, V_{DD}=V_{DD} Min
V_{OH}	Output High Voltage	0.9 V_{DD}		V	I_{OH}=-500 µA, V_{DD}=V_{DD} Min

T20.1 1237

1. I_{DD} active while a Read or Write (Program or Erase) operation is in progress.
2. The device is in Ready mode when no activity is on the LPC bus.
3. Do not violate processor or chipset specification regarding INIT# voltage.

TABLE 21: Recommended System Power-up Timings

Symbol	Parameter	Minimum	Units
$T_{PU-READ}$[1]	Power-up to Read Operation	100	µs
$T_{PU-WRITE}$[1]	Power-up to Write Operation	100	µs

T21.0 1237

1. This parameter is measured only for initial qualification and after a design or process change that could affect this parameter

TABLE 22: Pin Capacitance (V_{DD}=3.3V, T_A=25 °C, f=1 Mhz, other pins open)

Parameter	Description	Test Condition	Maximum
$C_{I/O}$[1]	I/O Pin Capacitance	$V_{I/O}$=0V	12 pF
C_{IN}[1]	Input Capacitance	V_{IN}=0V	12 pF
L_{PIN}[2]	Pin Inductance		20 nH

T22.0 1237

1. This parameter is measured only for initial qualification and after a design or process change that could affect this parameter.
2. Refer to PCI spec.

TABLE 23: Reliability Characteristics

Symbol	Parameter	Minimum Specification	Units	Test Method
N_{END}[1]	Endurance	10,000	Cycles	JEDEC Standard A117
T_{DR}[1]	Data Retention	100	Years	JEDEC Standard A103
I_{LTH}[1]	Latch Up	100 + I_{DD}	mA	JEDEC Standard 78

T23.0 1237

1. This parameter is measured only for initial qualification and after a design or process change that could affect this parameter.

TABLE 24: Clock Timing Parameters (LPC Mode)

Symbol	Parameter	33 MHz		66 MHz		Units
		Min	Max	Min	Max	
T_{CYC}	Cycle Time	30		15		ns
T_{HIGH}	LCLK High Time	11		6.5		ns
T_{LOW}	LCLK Low Time	11		6.5		ns
-	LCLK Slew Rate (peak-to-peak)	1	4	1	4	V/ns
-	RST# or INIT# Slew Rate	50		50		mV/ns

T24.0 1237

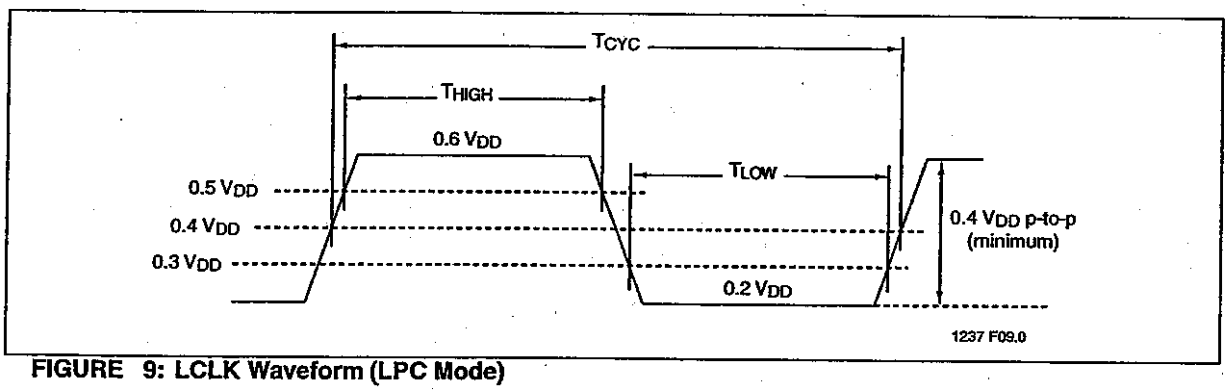


FIGURE 9: LCLK Waveform (LPC Mode)

TABLE 25: Reset Timing Parameters, V_{DD}=3.0-3.6V (LPC Mode)

Symbol	Parameter	Min	Max	Units
T_{PRST}	V_{DD} stable to Reset High	100		µs
T_{RSTP}	RST# Pulse Width	100		ns
T_{RSTF}	RST# Low to Output Float		48	ns
T_{RST}[1]	RST# High to LFRAME# Low	5		LCLK cycles
T_{RSTE}	RST# Low to reset during Sector-/Block-Erase or Program		10	µs

T25.0 1237

1. There will be a latency due to T_{RSTE} if a reset procedure is performed during a Program or Erase operation,

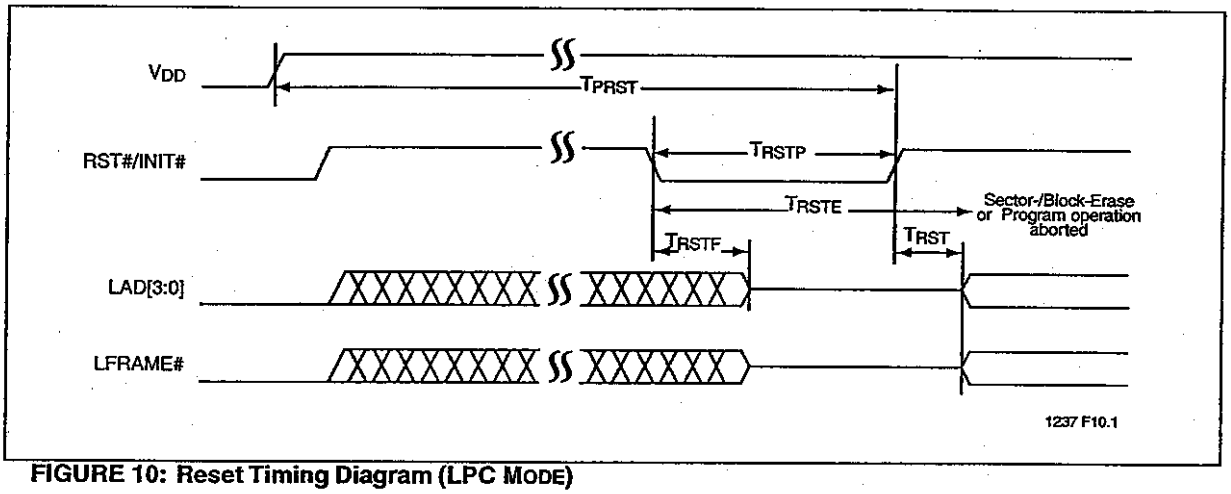


FIGURE 10: Reset Timing Diagram (LPC Mode)

AC Characteristics

TABLE 26: Read/Write Cycle Timing Parameters, V_{DD}=3.0-3.6V (LPC Mode)

Symbol	Parameter	33 MHz		66 MHz		Units
		Min	Max	Min	Max	
T_{CYC}	Clock Cycle Time	30		15		ns
T_{SU}	Data Set Up Time to Clock Rising	7		7		ns
T_{DH}	Clock Rising to Data Hold Time	0		0		ns
T_{VAL}[1]	Clock Rising to Data Valid	2	11	2	7	ns
T_{BP}	Byte Programming Time		10		10	µs
T_{SE}	Sector-Erase Time		25		25	ms
T_{BE}	Block-Erase Time		25		25	ms
T_{ES}	Program/Erase-Suspend Latency		10		10	µs
T_{ON}	Clock Rising to Active (Float to Active Delay)	2		2		ns
T_{OFF}	Clock Rising to Inactive (Active to Float Delay)		28		28	ns

T26.0 1237

1. Minimum and maximum times have different loads. See PCI spec

TABLE 27: AC Input/Output Specifications (LPC Mode)

Symbol	Parameter	Min	Max	Units	Conditions
I_{OH}(AC)	Switching Current High	-12 V_{DD}		mA	$0 < V_{OUT} \leq 0.3V_{DD}$
		-17.1(V_{DD}-V_{OUT})		mA	$0.3V_{DD} < V_{OUT} < 0.9V_{DD}$
			Equation C[1]		$0.7V_{DD} < V_{OUT} < V_{DD}$
	(Test Point)		-32 V_{DD}	mA	$V_{OUT} = 0.7V_{DD}$
I_{OL}(AC)	Switching Current Low	16 V_{DD}	Equation D[1]	mA	$V_{DD} > V_{OUT} \geq 0.6V_{DD}$
		26.7 V_{OUT}		mA	$0.6V_{DD} > V_{OUT} > 0.1V_{DD}$
					$0.18V_{DD} > V_{OUT} > 0$
	(Test Point)		38 V_{DD}	mA	$V_{OUT} = 0.18V_{DD}$
I_{CL}	Low Clamp Current	-25+(V_{IN}+1)/0.015		mA	$-3 < V_{IN} \leq -1$
I_{CH}	High Clamp Current	25+(V_{IN}-V_{DD}-1)/0.015		mA	$V_{DD}+4 > V_{IN} \geq V_{DD}+1$
slewr[2]	Output Rise Slew Rate	1	4	V/ns	$0.2V_{DD}$-$0.6V_{DD}$ load
slewf[2]	Output Fall Slew Rate	1	4	V/ns	$0.6V_{DD}$-$0.2V_{DD}$ load

T27.0 1237

1. See PCI spec.
2. PCI specification output load is used.

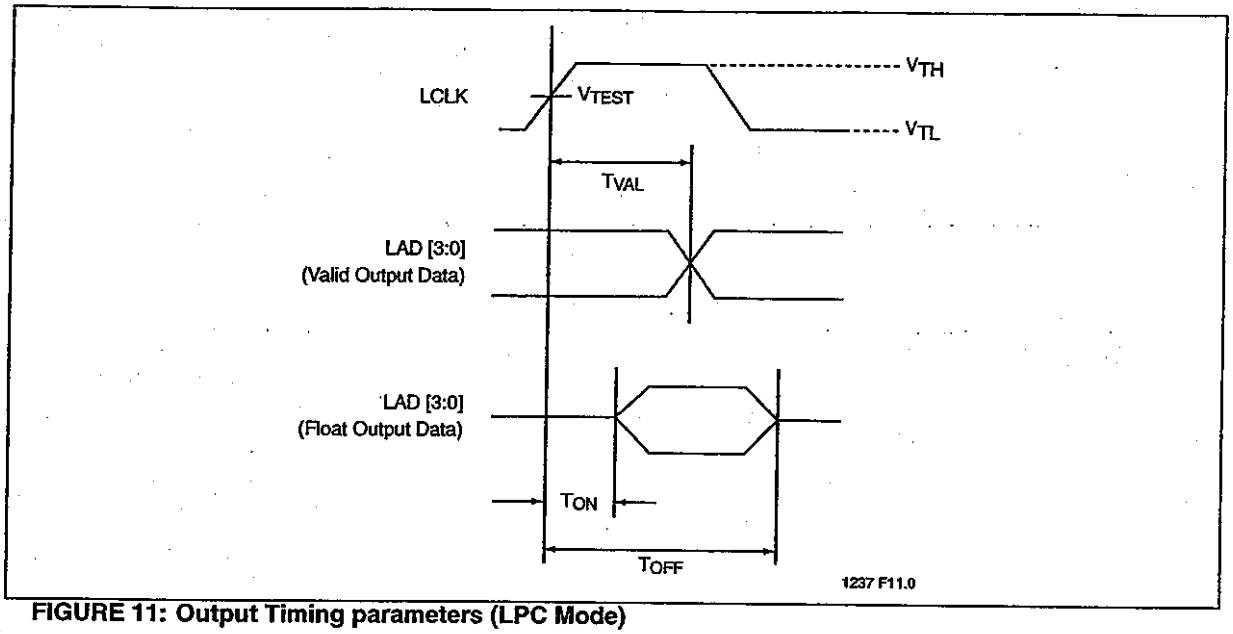


FIGURE 11: Output Timing parameters (LPC Mode)

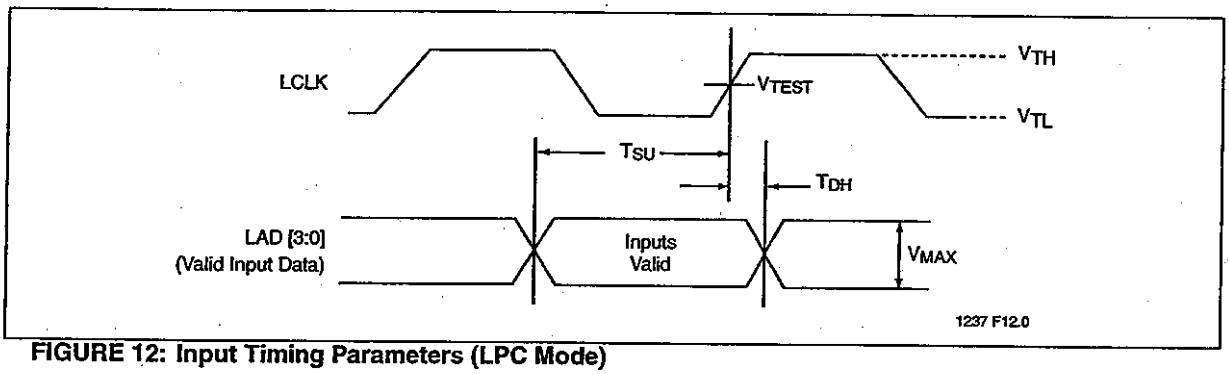


FIGURE 12: Input Timing Parameters (LPC Mode)

TABLE 28: Interface Measurement Condition Parameters (LPC Mode)

Symbol	Value	Units
V_{TH}[1]	0.6 V_{DD}	V
V_{TL}[1]	0.2 V_{DD}	V
V_{TEST}	0.4 V_{DD}	V
V_{MAX}[1]	0.4 V_{DD}	V
Input Signal Edge Rate	1	V/ns

T28.0 1237

1. The input test environment is done with 0.1 V_{DD} of overdrive over V_{IH} and V_{IL}. Timing parameters must be met with no more overdrive than this. V_{MAX} specifies the maximum peak-to-peak waveform allowed for measuring input timing. Production testing may use different voltage values, but must correlate results back to these parameters.

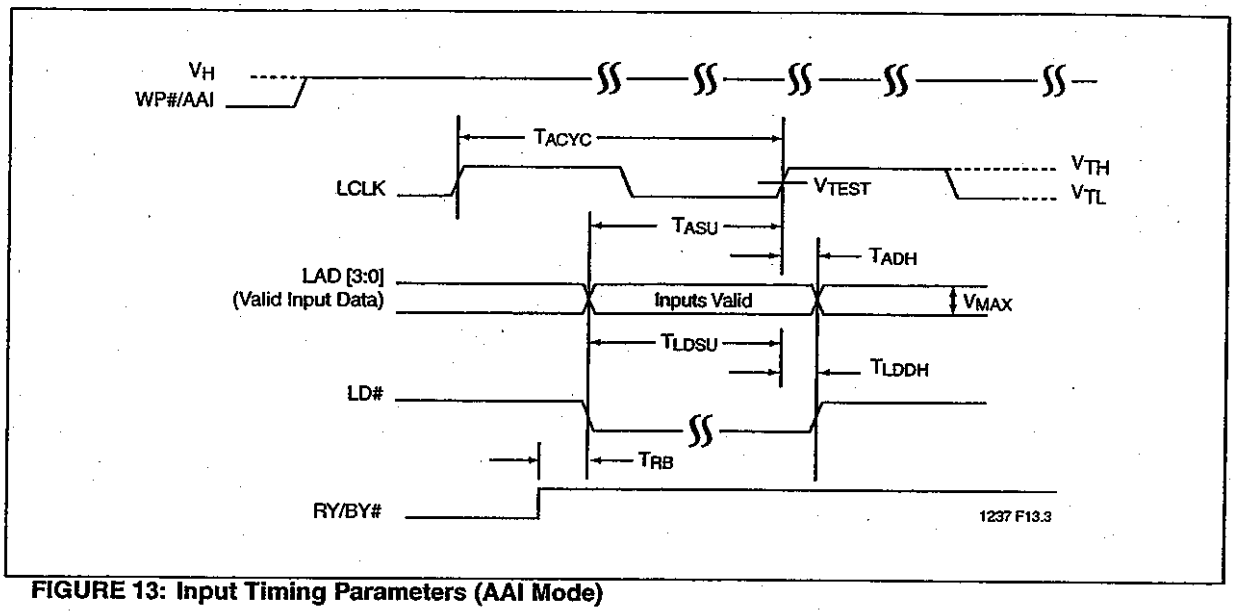
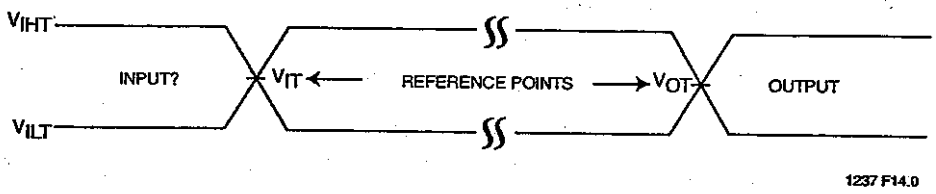


FIGURE 13: Input Timing Parameters (AAI Mode)

TABLE 29: Input Cycle Timing Parameters, V_{DD}=3.0-3.6V (AAI Mode)

Symbol	Parameter	Min	Max	Units
T_{ACYC}	Clock Cycle Time	135		ns
T_{ASU}	Data Set Up Time to Clock Rising	25		ns
T_{ADH}	Clock Rising to Data Hold Time	25		ns
T_{RB}	RY/BY# LD# Falling	25		ns
T_{LDSU}	LD# Set Up Time	25		ns
T_{LDDH}	LD# Hold Time	25		ns

T29.3 1237



AC test inputs are driven at V_{IHT} (0.9 V_{DD}) for a logic "1" and V_{ILT} (0.1 V_{DD}) for a logic "0". Measurement reference points for inputs and outputs are V_{IT} (0.5 V_{DD}) and V_{OT} (0.5 V_{DD}). Input rise and fall times (10% ↔ 90%) are <3 ns.

Note: V_{IT} - V_{INPUT} Test
V_{OT} - V_{OUTPUT} Test
V_{IHT} - V_{INPUT} HIGH Test
V_{ILT} - V_{INPUT} LOW Test

FIGURE 14: AC Input/Output Reference Waveforms

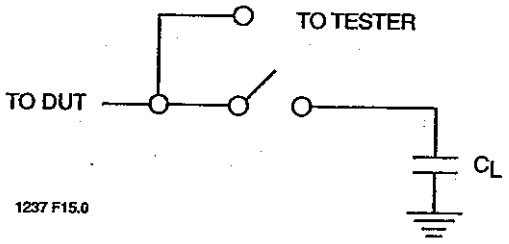
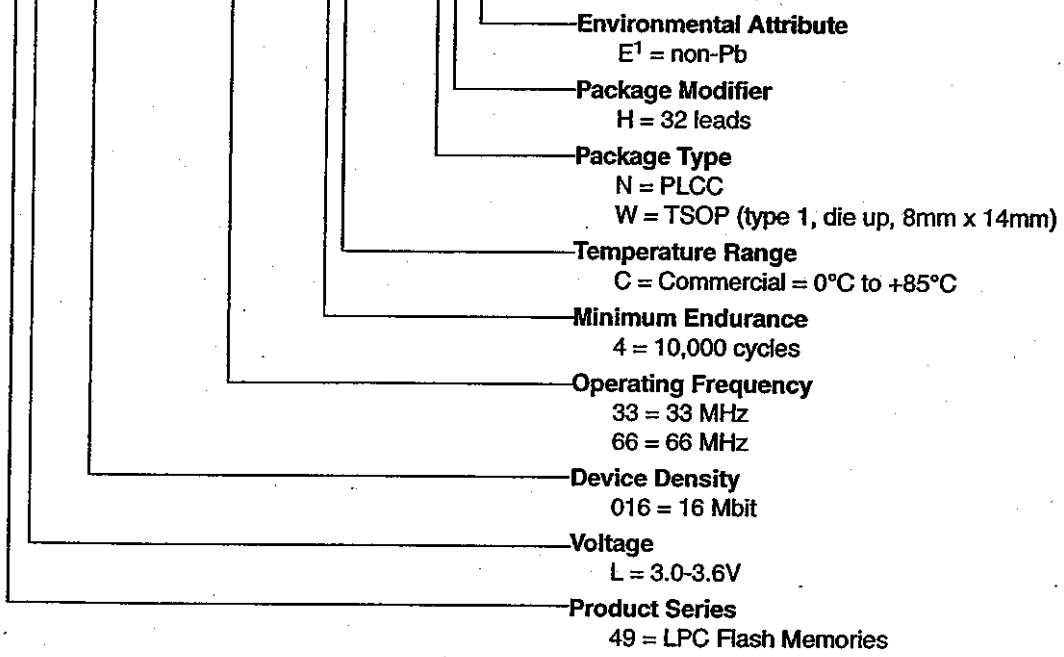


FIGURE 15: A Test Load Example

PRODUCT ORDERING INFORMATION

SST49LF 016C - XX - 4C - XH E
SST49LF xxxC - XX - XX - XX X



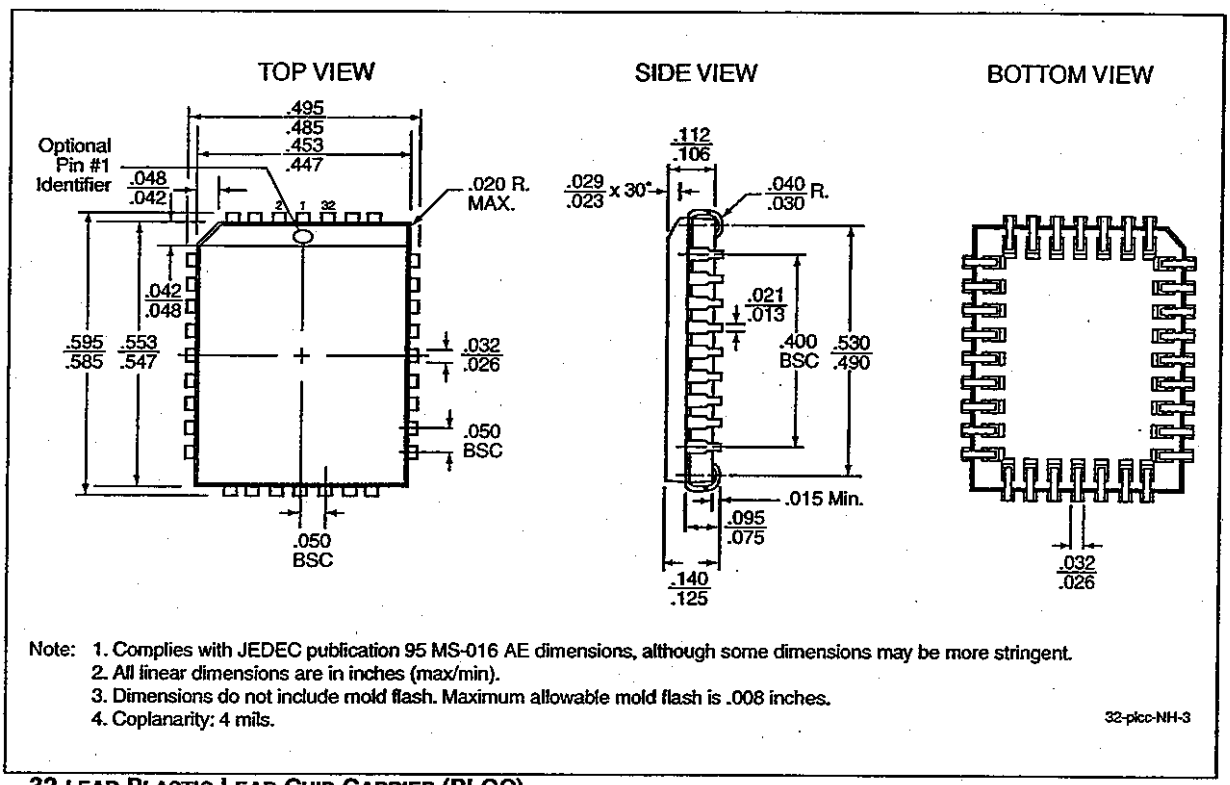
1. Environmental suffix "E" denotes non-Pb solder. SST non-Pb solder devices are "RoHS Compliant".

Valid combinations for SST49LF016C

SST49LF016C-33-4C-NHE SST49LF016C-33-4C-WHE
SST49LF016C-66-4C-NHE SST49LF016C-66-4C-WHE

Note: Valid combinations are those products in mass production or will be in mass production. Consult your SST sales representative to confirm availability of valid combinations and to determine availability of new combinations.

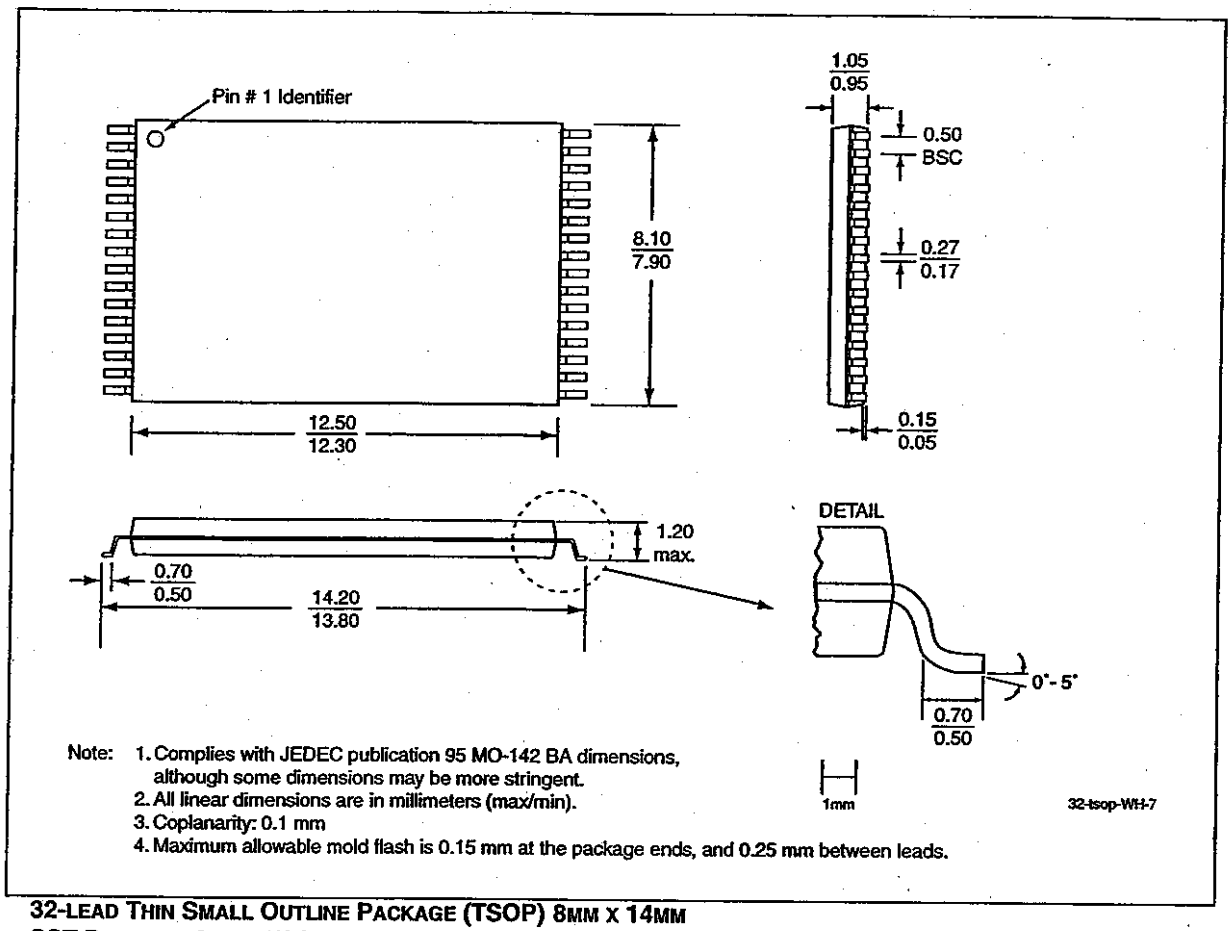
PACKAGING DIAGRAMS



Note: 1. Complies with JEDEC publication 95 MS-016 AE dimensions, although some dimensions may be more stringent.
2. All linear dimensions are in inches (max/min).
3. Dimensions do not include mold flash. Maximum allowable mold flash is .008 inches.
4. Coplanarity: 4 mils.

32-plcc-NH-3

32-LEAD PLASTIC LEAD CHIP CARRIER (PLCC)
SST PACKAGE CODE: NH



Note: 1. Complies with JEDEC publication 95 MO-142 BA dimensions, although some dimensions may be more stringent.
2. All linear dimensions are in millimeters (max/min).
3. Coplanarity: 0.1 mm
4. Maximum allowable mold flash is 0.15 mm at the package ends, and 0.25 mm between leads.

32-LEAD THIN SMALL OUTLINE PACKAGE (TSOP) 8MM X 14MM
SST PACKAGE CODE: WH

TABLE 30: Revision History

Revision	Description	Date
00	• S71237(01): Initial release of fact sheet (Advance Information)	Oct 2003
01	• S71237(01): Fact sheet changes • 2004 Flash Data Book	Nov 2003
02	• S71237(01): Fact sheet synchronized to and integrated into full data sheet • S71237: Initial release of data sheet (Advance Information) • Added Auto-Address Increment (AAI) mode	Apr 2004
03	• Added 32-TSOP (WH/WHE) package and associated MPNs • Clarified Supervoltage for AAI mode • Clarified the solder temperature profile under "Absolute Maximum Stress Ratings" on page 26	Dec 2004
04	• Obsoleted stand-alone Fact Sheet S71237(01) • Changed to firmware protocol-only data sheet • Removed the EI package and related MPNs • Added RoHS compliance information on page 1 and in the "Product Ordering Information" on page 34 • Updated the surface mount lead temperature from 240°C to 260°C and the time from 3 seconds to 10 seconds on page 26.	Jul 2005
05	• Removed leaded part numbers	Jan 2006
06	• Cosmetic update to Figure 3	Feb 2006
07	• Revised Product Description. Added 66 MHz information to Features. Added 66 MHz values to Table 24, and Table 26. Added "33 MHz and 66 MHz" to Table title and removed f=33 MHz from table cells in Table 20.	Sep 2006

EXHIBIT C

Intel® Serial Flash Memory (S33)

16-, 32-, and 64-Mbit

Datasheet

Product Features

- Architecture
  - SPI-compatible serial interface
  - Eight 8-Kbyte parameter blocks; configurable as one 64-Kbyte main memory sector
  - 64-Kbyte main memory sectors
    - 16 Mbit (31 sectors)
    - 32 Mbit (63 sectors)
    - 64 Mbit (127 sectors)
- Voltage and Power
  - Vcc = 2.7 V to 3.6 V
  - Standby current: 15 µA (Typ)
  - Read current: 3 mA (Typ) at 33.3 MHz; 6 mA (Typ) at 68 MHz
- Performance
  - 68-MHz fast read; 33.3 MHz standard read
  - 256-byte program buffer
- Software
  - Driver and file manager
- Security
  - One-Time Programmable Space
    - 64 unique factory device identifier bits
    - 64 user-programmable OTP bits
    - Additional 3920 user-programmable OTP bits
- Quality and Reliability
  - Operating temperature: -40 °C to +85 °C
  - 100K minimum erase cycles per sector (block)
  - 0.13 µm ETOX™ VIII process
- Density and Packaging
  - 16, -32, -64-Mbit densities in SOIC-16 package
  - 16-Mbit density in SOIC-8 package
  - Industry standard packaging and pinout

Order Number: 314822-001US
August 2006

Contents

Figures

Tables

Revision History

Date	Revision	Description
August 2006	001	Initial release

1.0 Introduction

This document describes the Intel® Serial Flash Memory (S33) device features, operation, and specifications.

The S33 device provides superior performance with enhanced security features, taking advantage of the high quality and reliability of the NOR-based Intel® 0.13 µm ETOX™ VIII process technology. Offered in 16-Mbit, 32-Mbit, and 64-Mbit densities, the S33 device is hardware and software compatible with existing industry offerings to ensure ease of design.

The S33 device takes advantage of more than one billion units of flash manufacturing experience and is ideal for code and data applications where simplified interface and low cost are the primary requirements. Examples include PCs and notebooks, WLAN and DSL cards and routers, printers, TVs, DVD/CD players and recorders, and other consumer electronic devices.

1.1 Nomenclature

Block:	A group of flash cells that share common erase circuitry and erase simultaneously. For Serial Flash devices, Blocks are more commonly referred to as Sectors.
Cleared:	Indicates a logic zero (0).
Page:	A 256-byte main memory segment that is aligned to a 256-byte boundary.
Parameter Block:	An 8-KB memory segment that can be erased independently.
Program:	A data write operation to the flash array.
Sector:	A single 64-KB main block, or all eight of the 8-KB parameter blocks combined.
Set:	Indicates a logic one (1).

1.2 Acronyms

LSB:	Least Significant Bit
MSB:	Most Significant Bit
OTP:	One-Time Programmable
PR:	Protection Register
PRD:	Protection Register Data
RFU:	Reserved for Future Use
SR:	Status Register
WEL:	Write Enable Latch
WSM:	Write State Machine

1.3 Conventions

0x:	Hexadecimal prefix
0b:	Binary prefix
K:	1,000
M:	1,000,000
Nibble:	4 bits
Byte:	8 bits
Word:	16 bits
Kword:	1,024 words
Kb:	1,024 bits
KB:	1,024 bytes
Mb:	1,048,576 bits
MB:	1,048,576 bytes
Brackets:	Square brackets ([]) will be used to designate group membership or to define a group of signals with similar function (i.e. A[21:1], SR[4,1] and D[15:0]).
00FFh:	Denotes 16-bit hexadecimal numbers
00FF 00FFh:	Denotes 32-bit hexadecimal numbers

2.0 Functional Overview

This section provides an overview of the features and capabilities of the Intel® Serial Flash Memory (S33) device.

The S33 device is available in 16-, 32-, and 64-Mbit densities with a common SPI interface. The SPI interface consists of 8 pins. Six of these pins are signals; the other two are Vcc and ground.

The S33 device contains eight 8-Kbyte parameter blocks and up to 127 64-Kbyte main memory sectors. The eight 8-Kbyte parameter blocks can be treated as one 64-Kbyte main memory sector.

The S33 device includes new security features. Two 8-byte, thirty 16-byte, and one 10-byte individually-lockable OTP Protection Registers can support multiple uses, including unique flash device identification.

A sector erase operation erases one of the device's 64-Kbyte main memory sectors or all eight 8-Kbyte parameter blocks in parallel, independent of other memory sectors. A parameter block erase operation erases the specified parameter block; the operation is ignored when the address is outside the parameter block space. Each block or memory sector can be independently erased 100,000 times. A bulk erase operation erases the entire chip.

Each device incorporates a write buffer of 256 bytes (128 words) to allow optimum programming performance. Page program operation uses the write buffer to program up to 256-bytes within a 256-byte-aligned main memory region. A given page can be programmed multiple times between erase cycles.

2.1 Memory Maps

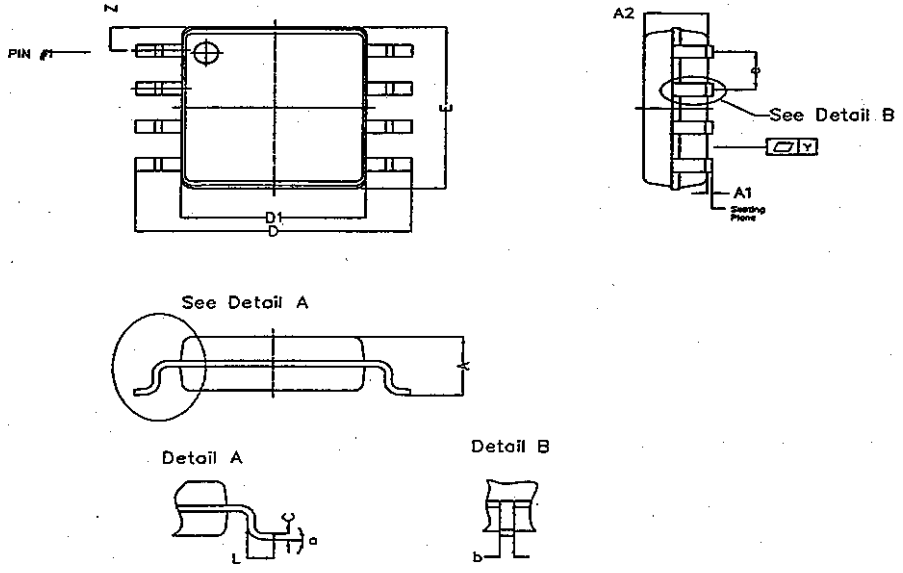
Table 1. Bottom Boot Memory Map

Size (KB)	Sector	Memory Addressing (Hex)	Memory Addressing (Hex)	Memory Addressing (Hex)
		64-Mbit	32-Mbit	16-Mbit
64	127	7F0000 - 7FFFFF		
64	126	7E0000 - 7EFFFF		
...	...	...		
64	64	400000 - 40FFFF		
64	63	3F0000 - 3FFFFF	3F0000 - 3FFFFF	
64	62	3E0000 - 3EFFFF	3E0000 - 3EFFFF	
...	...	...	...	
64	32	200000 - 20FFFF	200000 - 20FFFF	
64	31	1F0000- 1FFFFF	1F0000- 1FFFFF	1F0000- 1FFFFF
64	30	1E0000 - 1EFFFF	1E0000 - 1EFFFF	1E0000 - 1EFFFF
...	...	...	...	...
64	16	100000 - 10FFFF	100000 - 10FFFF	100000 - 10FFFF
64	15	F0000 - FFFFF	F0000 - FFFFF	F0000 - FFFFF
64	14	E0000 - EFFFF	E0000 - EFFFF	E0000 - EFFFF
...	...	...	...	...
64	4	40000 - 4FFFF	40000 - 4FFFF	40000 - 4FFFF
64	3	30000 - 3FFFF	30000 - 3FFFF	30000 - 3FFFF
64	2	20000 - 2FFFF	20000 - 2FFFF	20000 - 2FFFF
64	1	10000 - 1FFFF	10000 - 1FFFF	10000 - 1FFFF
8	0-H	E000 - FFFF	E000 - FFFF	E000 - FFFF
8	0-G	C000 - DFFF	C000 - DFFF	C000 - DFFF
8	0-F	A000 - BFFF	A000 - BFFF	A000 - BFFF
8	0-E	8000 - 9FFF	8000 - 9FFF	8000 - 9FFF
8	0-D	6000 - 7FFF	6000 - 7FFF	6000 - 7FFF
8	0-C	4000 - 5FFF	4000 - 5FFF	4000 - 5FFF
8	0-B	2000 - 3FFF	2000 - 3FFF	2000 - 3FFF
8	0-A	0 - 1FFF	0 - 1FFF	0 - 1FFF

3.0 Package Information

3.1 SOIC-8 Package

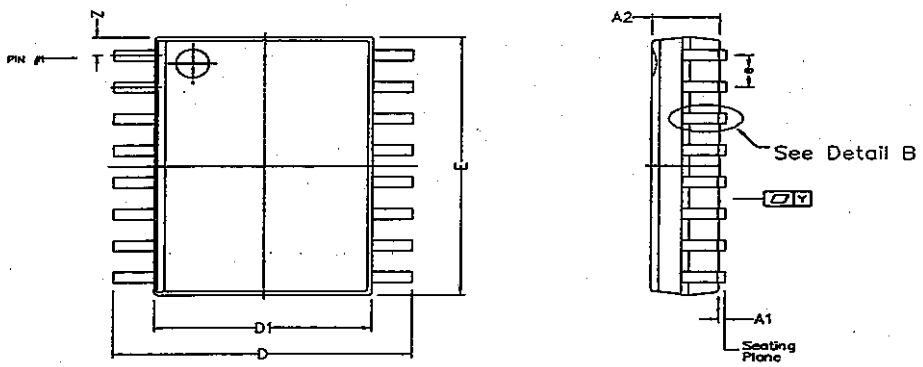
Figure 1. SOIC-8 Package Drawing and Specifications



Product Information	Symbol	Millimeters			Inches		
		Min	Nom	Max	Min	Nom	Max
Package Height	A	-	-	2.159	-	-	0.085
Standoff	A1	0.051	-	-	0.002	-	-
Package Body Thickness	A2	1.702	1.803	1.905	0.067	0.071	0.075
Lead Width	b	0.356	0.419	0.483	0.014	0.017	0.019
Lead Thickness	c	0.191	0.216	0.241	0.008	0.009	0.010
Package Body Length	D1	5.131	5.232	5.334	0.202	0.206	0.210
Package Body Width	E	5.131	5.232	5.334	0.202	0.206	0.210
Lead Pitch	e	-	1.270	-	-	0.050	-
Terminal Dimension	D	7.762	7.912	8.062	0.306	0.312	0.318
Lead Tip Length	L	0.508	0.635	0.762	0.020	0.025	0.030
Lead Count	N	-	8	-	-	8	-
Lead Tip Angle	a	0°	4°	8°	0°	4°	8°
Seating Plane Coplanarity	Y	-	-	0.100	-	-	0.004

3.2 SOIC-16 Package

Figure 2. SOIC-16 Package Drawing and Specifications



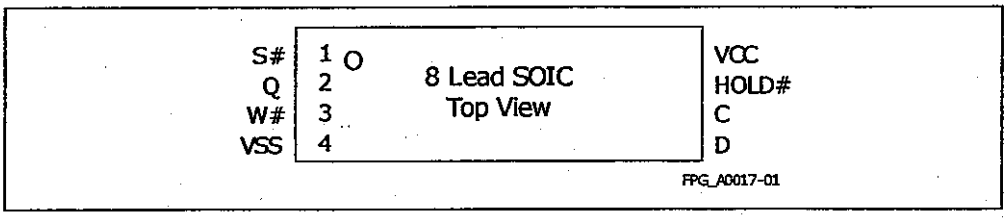
Product Information	Symbol	Millimeters			Inches		
		Min	Nom	Max	Min	Nom	Max
Package Height	A	-	-	2.642	-	-	0.104
Standoff	A1	0.102	-	-	0.004	-	-
Package Body Thickness	A2	2.286	2.362	2.438	0.090	0.093	0.096
Lead Width	b	0.315	0.412	0.508	0.012	0.016	0.020
Lead Thickness	c	0.231	0.274	0.318	0.009	0.011	0.013
Package Body Length	D1	7.442	7.519	7.595	0.293	0.296	0.299
Package Body Width	E	10.262	10.338	10.414	0.404	0.407	0.410
Lead Pitch	e	-	1.270	-	-	0.050	-
Terminal Dimension	D	10.190	10.340	10.490	0.401	0.407	0.413
Lead Tip Length	L	0.406	0.648	0.889	0.016	0.026	0.035
Lead Count	N	-	16	-	-	16	-
Lead Tip Angle	α	0°	4°	8°	0°	4°	8°
Seating Plane Coplanarity	Y	-	-	0.100	-	-	0.004

4.0 Pinouts and Signal Descriptions

The Intel® Serial Flash Memory (S33) device is available in two package types. The 16-Mbit density is supported with SOIC-8 and SOIC-16 packages as shown in Figure 3 and Figure 4. The 32- and 64-Mbit densities are supported by the SOIC-16 package as shown in Figure 4.

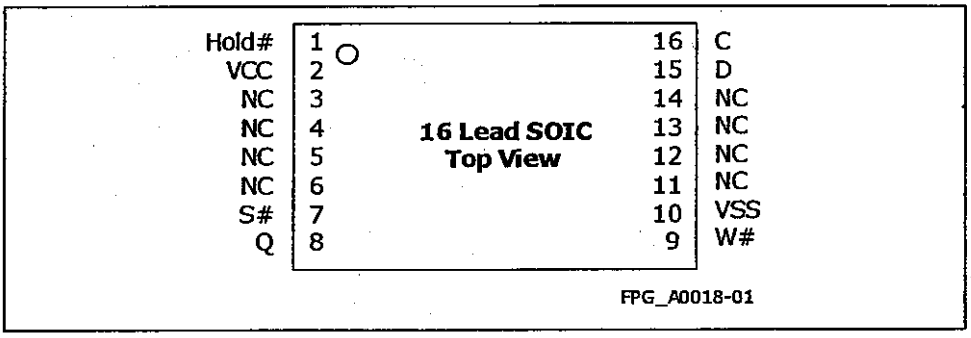
4.1 SOIC-8 Pinout (16-Mbit)

Figure 3. SOIC-8 Pinout



4.2 SOIC-16 Package Pinout

Figure 4. SOIC-16 Package Pinout



4.3 Signal Descriptions

Table 2. Signal Descriptions

Symbol	Type	Name and Function
C	Input	**SPI Clock:** Provides the timing of the SPI interface. OP codes, addresses, and data are latched in on the rising edge. SPI output data transitions after the falling edge.
D	Input	**SPI Data Input:** Shifts all data (including OP codes, Address Bytes, as well as Data Bytes) into the device. All data is clocked in on the rising edge of "C", starting with the MSB. The rising edge input applies to Modes 0 & 3 as depicted in Figure 12, "Supported SPI Bus Operation Modes" on page 25.
Q	Output	**SPI Data Output:** Shifts all data out of the device. All output data is clocked out after the falling edge of "C", starting with the MSB. The falling edge output applies to Modes 0 & 3 as depicted in Figure 12 on page 25.
S#	Input	**SPI Select:** Falling S# edge triggers command writes to the SPI interface. Rising S# edge completes (or terminates) the SPI command cycle. When S# is high, "Q" is at high-Z.
HOLD#	Input	**SPI HOLD:** Internally freezes the Synchronization Clock and sets "Q" to high-Z. To enter the Hold condition, S# must be low. Refer to Section 8.1.2, "The Hold State" on page 25 for details.
W#	Input	**Write Protect:** Enables write protection. Refer to Table 18 on page 33 for details.
VCC	Power	**Power Supply:** Source voltage. Writes to the flash array are inhibited when $V_{CC} \leq V_{LKO}$. Operations at invalid V_{CC} should not be attempted.
VSS	Power	**Ground:** Connect to system ground. Do not float VSS.

5.0 Maximum Ratings and Operating Conditions

5.1 Absolute Maximum Ratings

Warning: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only.

> ***Note:*** This document contains information available at the time of its release. The specifications are subject to change without notice. Verify with your local Intel sales office that you have the latest datasheet before finalizing a design.

Table 3. Absolute Maximum Ratings

Parameter	Min	Max	Unit	Notes
Temperature under Bias Expanded (T_A, Ambient)	–40	+85	°C	4
Storage Temperature	–65	+125	°C	—
V_{CC} Voltage	–2.0	+5.6	V	1, 2
I_{SH} Output Short Circuit Current	—	100	mA	3

Notes:
1. Voltage is referenced to V_{SS}. During infrequent non-periodic transitions, the voltage potential between V_{SS} and input/output pins may undershoot to –2.0 V for periods < 20 ns or overshoot to V_{CCQ} (max) + 2.0 V for periods < 20 ns.
2. During infrequent non-periodic transitions, the voltage potential between V_{CC} and the supplies may undershoot to –2.0 V for periods < 20 ns or V_{SUPPLY} (max) + 2.0 V for periods < 20 ns.
3. Output is shorted for no more than one second. No more than one output shorted at a time
4. Temperature specified is at ambient (T_A) and not the package (T_C).

5.2 Operating Conditions

Warning: Operation beyond the "Operating Conditions" is not recommended. Extended exposure beyond the "Operating Conditions" may affect device reliability.

Table 5. Temperature and VCC Operating Condition

Symbol	Parameter	Min	Nom	Max	Unit	Notes
T_C	Operating Temperature (case)	–40	+25	+85	°C	1
V_{CC}	V_{CC} Supply Voltage	2.7	3.0	3.6	V	

Note:
1. Temperature specified is at case (T_C) and/or ambient (T_A).

5.3 Power Supply

This section provides an overview of system-level considerations with regards to the flash device. It includes a brief description of power-up, power-down and decoupling design considerations.

5.3.1 Power-Up/Down Characteristics

The device is protected against accidental block erasure or programming during power transitions. After power-up, a t_{VSL} latency is required before S# can be brought low to issue a command.

5.3.2 Power Supply Decoupling

Flash memory devices require careful power supply decoupling. The basic power supply considerations are as follows:

1. Standby current levels
2. Active current levels
3. Transient peaks produced when the device is enabled

When the device is accessed, many internal conditions change. Circuits within the device enable charge-pumps, and internal logic states change at high speed. All of these internal activities produce transient signals. Transient current magnitudes depend on the capacitive and inductive loading at the device output. Two-line control and correct de-coupling capacitor selection suppress transient voltage peaks.

Additionally, for every eight devices used in the system, a 4.7 µF electrolytic capacitor should be placed between power and ground close to the devices. The bulk capacitor is meant to overcome voltage droop caused by PCB trace inductance.

6.0 Electrical Characteristics

6.1 DC Current Specifications

Table 6. DC Current Characteristics

Symbol	Parameter	Typ	Max	Unit	Test Conditions	Notes
I_{LI}	Input Load Current	—	±2	µA	$V_{CC} = V_{CC}$ Max; $V_{IN} = V_{CC}$ or V_{SS}	3
I_{LO}	Output Leakage Current	—	±2	µA	$V_{CC} = V_{CC}$ Max; $V_{IN} = V_{CC}$ or V_{SS}	—
I_{CCS}	V_{CC} Standby Current	15	70	µA	S# = V_{CC}; $V_{IN} = V_{CC}$ or V_{SS}	4,6
I_{DPD}	V_{CC} Deep Power-Down Current	15	70	µA	S# = V_{CC}; $V_{IN} = V_{CC}$ or V_{SS}	4
I_{CCR}	V_{CC} Read Current	2	4	mA	20 MHz; No Load	—
		3	5		33.3 MHz; No Load	7
		5	8		50 MHz; No Load	
		6	10		68 MHz; No Load	
I_{CCP}	Page Program Current	18	25	mA	S# = V_{CC}; 256-Byte Page	2,5
I_{CCE}	Parameter Block Erase Current	16	45	mA	S# = V_{CC}	2.5
	Sector Erase Current	16	45	mA	S# = V_{CC}	2,5
I_{CCB}	Bulk Erase Current	16	45	mA	S# = V_{CC}	2.5

Notes:
1. All currents are RMS unless noted. Typical values at typical V_{CC}, T_C = +25 °C.
2. Sampled, not 100% tested.
3. If $V_{IN} > V_{CC}$ the input load current increases to 10 µA max.
4. I_{CCS} and I_{DPD} is the average current measured over any 5 ms time interval 5 µs after a S# de-assertion.
5. I_{CCP}, I_{CCE} measured over typical or max times.
6. I_{CCS} will increase substantially if W# or HOLD# is toggled while in standby mode.
7. I_{CCR} will increase if D is toggled during read.

6.2 DC Voltage Specifications

Table 7. DC Voltage Characteristics

Symbol	Parameter	Min	Max	Unit	Test Conditions	Notes
V_{IL}	Input Low Voltage	0	0.3 * V_{CC}	V	—	1
V_{IH}	Input High Voltage	0.7 * V_{CC}	V_{CC}	V	—	1
V_{OL}	Output Low Voltage	—	0.1	V	$V_{CC} = V_{CC}$Min I_{OL} = 100 µA	
V_{OH}	Output High Voltage	V_{CC} – 0.1	—	V	$V_{CC} = V_{CC}$Min I_{OH} = –100 µA	—
V_{LKO}	V_{CC} Lockout Voltage	2.0	—	V	—	—

Note:
1. V_{IL} can undershoot to –1.0 V for periods <2 ns and V_{IH} may overshoot to a maximum of V_{CC} +1.0 V for periods < 2 ns.

6.3 Capacitance

Table 8. Device Capacitance

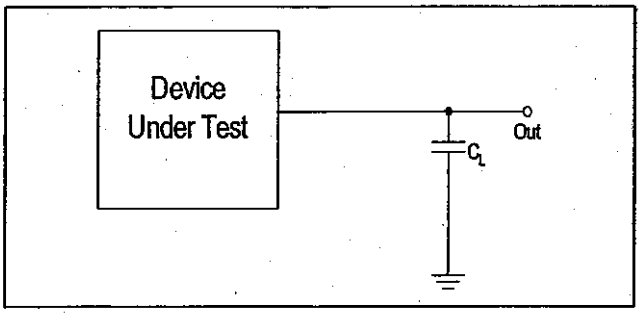
Symbol	Parameter[1]	Type	Max	Unit	Condition[2]
C_{IN}	Input Capacitance	6	8	pF	V_{IN} = 0.0 V V_{CC} = 0 V "or" V_{CC} = V_{CC}max
C_{OUT}	Output Capacitance	8	12	pF	V_{OUT} = 0.0 V V_{CC} = 0 V "or" V_{CC} = V_{CC}max

Notes:
1. Sampled, not 100% tested.
2. T_A = +25 °C, f = 1 MHz.

7.0 AC Characteristics

7.1 AC Test Conditions

Figure 5. Transient Equivalent Testing Load Circuit



Note: C_L Includes Jig Capacitance

Table 9. AC Measurement Conditions

Parameter	Min	Max	Unit
Load Capacitance (C_L)	—	30[1]	pF
Input Rise and Fall Times	0.3	3	ns
Input V^{IL} Timing Reference Voltages	—	0.3*Vcc	V
Input V^{IH} Timing Reference Voltages	0.7*Vcc	—	V
Input Drive Voltages	0 for V_{IL}, Vcc for V_{IH}		V
Input Timing Reference Voltages	0.3 V_{CC} to 0.7 V_{CC}		V
Output Timing Reference Voltages	$V_{CC}/2$		V

Note:
1. Output Hi-Z is defined as the point where data output is no longer driven.

7.2 AC Timing Characteristics

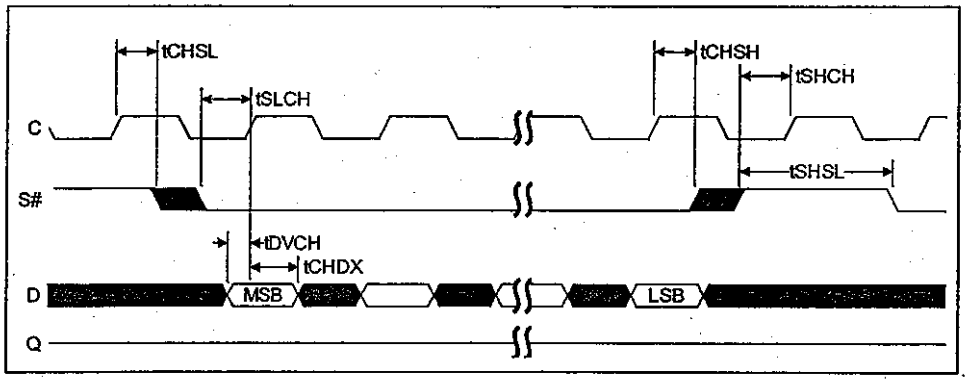
7.2.1 Serial Input Characteristics

Table 10. Serial Input Characteristics

Sym	Parameter	Min	Max	Unit	Notes
f_C	Clock Frequency for all instructions except READ	D.C.	68.0	MHz	—
f_R	Clock Frequency for READ	D.C.	33.3	MHz	—
t_{CH}	Clock High Time	7	—	ns	1
t_{CL}	Clock Low Time	7	—	ns	1
t_{CLCH}	Clock Rise Time (peak to peak)	0.1	—	V/ns	2, 3,4
t_{CHCL}	Clock Fall Time (peak to peak)	0.1	—	V/ns	2, 3,4
t_{CHSL}	S# Active Setup Time (relative to preceding C edge)	5	—	ns	—
t_{SLCH}	S# Active Setup Time (relative to subsequent C edge)	5	—	ns	—
t_{CHSH}	S# Inactive Hold Time (relative to C)	5	—	ns	—
t_{SHCH}	S# Inactive Setup Time (relative to C)	5	—	ns	—
t_{SHSL}	S# Deselect Time	100	—	ns	—
t_{DVCH}	Data Input Setup Time	2	—	ns	—
t_{CHDX}	Data Input Hold Time	5	—	ns	—

Notes:
1. $t_{CH} + t_{CL}$ must be greater than or equal to $1/f_C$(max).
2. Sampled, not 100% tested.
3. Expressed as a slew-rate.
† Minimum clock rise/fall times guarantee functionality. Clock rise/fall times must fall within the range specified in Figure 9, "AC Measurement Conditions" on page 19 for compliance to timing specs.

Figure 6. Serial Input Timing



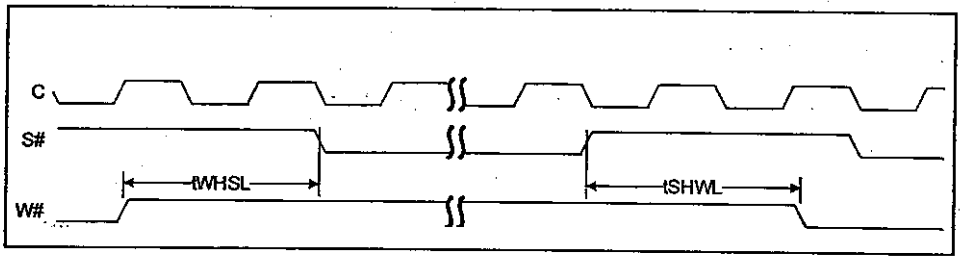
7.2.2 Write Protect Setup and Hold Timing

Table 11. Write Protect Setup and Hold Timing

Sym	Parameter	Min	Max	Unit	Notes
t_{WHSL}	W# Setup Time	20	—	ns	1
t_{SHWL}	W# Hold Time	100	—	ns	1

Note:
1. Only applicable as a constraint for a WRSR instruction when SRWD is set to 1.

Figure 7. Write Protect Setup and Hold Timing during WRSR when SRWD=1



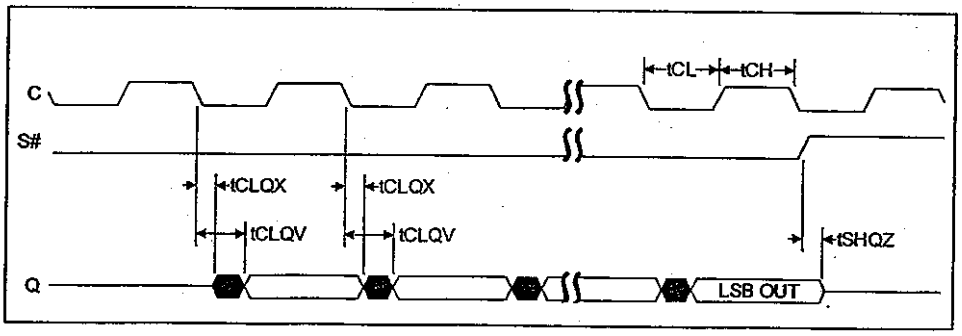
7.2.3 Output Timing

Table 12. Output Timing

Sym	Parameter	Min	Max	Unit	Notes
t_{CLQV}	Clock Low to Output Valid (30 pF, 2.7 V to 3.6 V)	—	8	ns	—
t_{CLQV}	Clock Low to Output Valid (10 pF, 3.0 V to 3.6 V)	—	6	ns	—
t_{CLQX}	Output Hold Time	0	—	ns	—
t_{SHQZ}	Output Disable Time	—	8	ns	1

Note:
1. Sampled, not 100% tested.

Figure 8. Output Timing



7.2.4 Hold Timing

Table 13. Hold Timing

Sym	Parameter	Min	Max	Unit	Notes
t_{CHHL}	HOLD# Assertion Hold Time (relative to C)	5	—	ns	—
t_{HLCH}	HOLD# Assertion Setup Time (relative to C)	5	—	ns	—
t_{HLQZ}	HOLD# Assertion to Output High-Z	—	8	ns	1
t_{CHHH}	HOLD# De-assertion Hold Time (relative to C)	5	—	ns	—
t_{HHCH}	HOLD# De-assertion Setup Time (relative to C)	5	—	ns	—
t_{HHQX}	HOLD# De-assertion to Output Low-Z	—	8	ns	1

Note:
1. Sampled, not 100% tested.

Figure 9. Hold Timing - Standard Usage

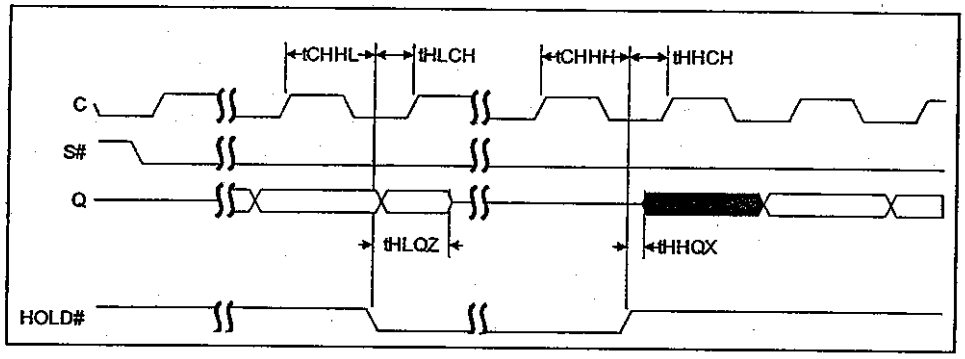
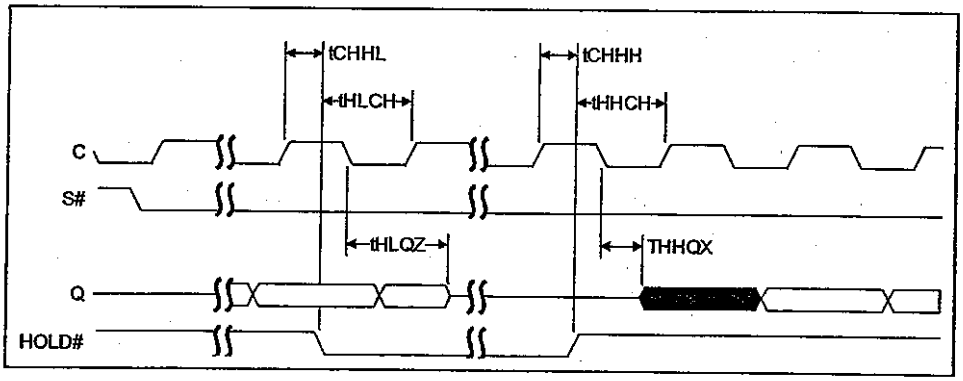


Figure 10. Hold Timing - Non-standard Usage

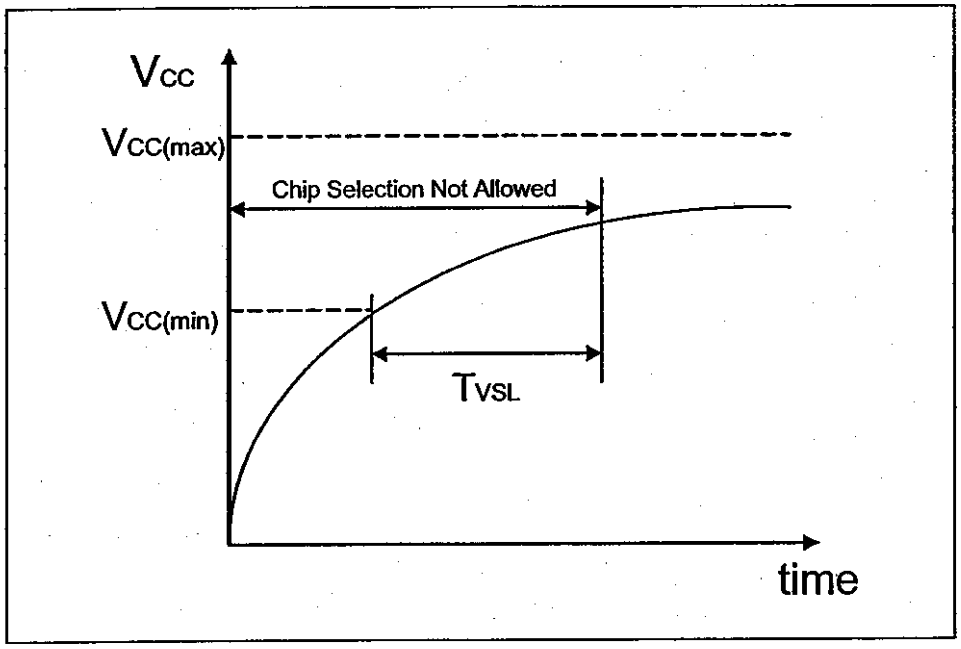


7.2.5 Other Timings

Table 14. Other Timings

Sym	Parameter	Min	Typ	Max	Unit	Notes
t_{PP}	Page Program Time	—	1.4	10	ms	1
t_{BP}	Byte OTP Program Time	—	40	175	µs	1
t_{SE}	Parameter Block Erase Time (8 KB)	—	0.3	2.5	s	1
	Sector Erase Time (64 KB)	—	0.7	4.0	s	1
t_{BE}	Bulk Erase Time (64 M)	—	89.6	512	s	1
	Bulk Erase Time (32 M)	—	44.8	256	s	1
	Bulk Erase Time (16 M)	—	22.4	128	s	1
t_{VSL}	V_{CC} power valid to S# assertion (low)	60	—	—	µs	—
t_{RDP}	Release from DPD mode into standby mode	60	—	—	µs	—

Figure 11. T_{VSL} Power-up Timing



8.0 Device Operations

This section provides an overview of the Intel® Serial Flash Memory (S33) device operations.

8.1 SPI Bus Operations

The SPI instruction cycle begins with a byte-wide OP Code that is initiated with the falling edge of S#. The 8-bit instruction is latched into "D" (data input), MSB first, on the rising edge of "C" (clock).

Some OP Codes are followed by an additional address and dummy and/or data bytes, MSB first. The number of input instruction bytes depends upon the OP Code. Refer to Table 15, "SPI Command Set" on page 26 for the instruction protocols. Address and dummy bytes are input through "D" on the rising "C" edge.

Depending upon the OP Code, the data bytes are either *input* data through "D", or they are *output* data from "Q". On cycles that input data through "D", the output signal "Q" is at high-Z.

For instructions that change the memory contents or device configuration (such as a Status Register Write command), the rising S# edge must occur on a whole-byte increment, otherwise the command will be ignored.

For read operations, the instruction sequence can be botched (ignored) only if S# is raised before the input sequence is complete. After the required number of input bits is clocked into the device, a data stream is clocked out of "Q"; each bit is shifted out after the falling edge of "C" (MSB first). When data is streaming from "Q", raising S# will terminate the data stream and bring this output to high-Z.

The rising S# edge always resets the SPI command interpreter and places the output in high-Z. It also does one of the following actions:

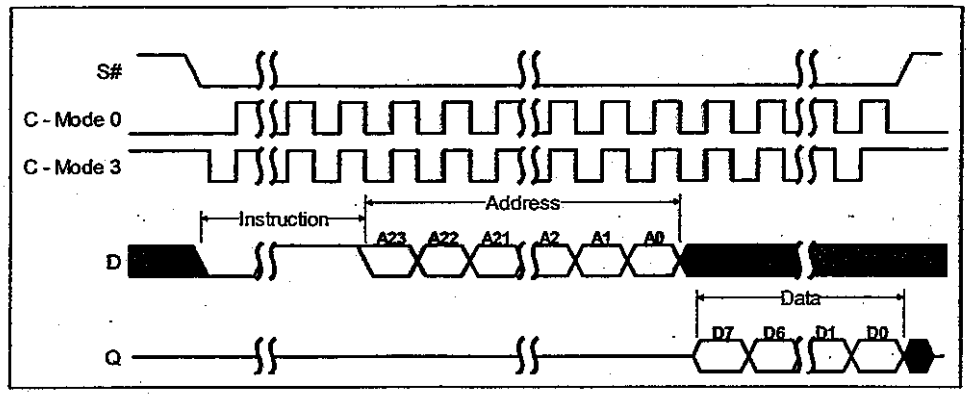
- Terminates the output data stream (read operations)
- Kicks off program/erase algorithms
- Initiates changes to the SR
- Botches an SPI command when S# is raised too early (or too late for commands that alter the array or device configuration)
- Terminates a command and puts the device in standby mode (not in the case of a program or erase operation)

When S# is high and the internal algorithms are completed, the device will go into standby mode.

8.1.1 SPI Modes

This device supports SPI bus operations Mode 0 and Mode 3, as depicted in Figure 12, "Supported SPI Bus Operation Modes" on page 25. The difference between the two modes is the default state of the clock signal (C) when the SPI bus master is in standby. For Mode 0, the "C" is normally low; for Mode 3, "C" is normally high. For both modes, input data (D) is sampled on the rising edge of "C", and output data (Q) is updated on the falling edge of "C."

Figure 12. Supported SPI Bus Operation Modes



8.1.2 The Hold State

The HOLD# input signal freezes the internal Clock (C) without resetting the device's clocking sequence. However, taking HOLD# to V_{IL} does not terminate any program or erase operation that is currently in progress.

To enter the Hold State, the device must be selected (S# at V_{IL}). The Hold State starts on the falling edge of the HOLD# signal, provided that it coincides with the Clock at V_{IL} as shown in Figure 13 on page 25. The Hold State ends on the rising edge of HOLD#, provided that it coincides with the SPI Clock at V_{IL}.

If the falling HOLD# edge does not coincide with the Clock at V_{IL}, the Hold State starts after the next falling edge of the Clock as shown in Figure 14 on page 26. Similarly, if the rising edge does not coincide with the Clock at V_{IL}, the Hold State ends after the next falling edge of the Clock.

The command sequence will not necessarily botch if S# is raised to V_{IH} while HOLD# is at V_{IL}. Raising S# will complete the command sequence, regardless of the state of HOLD#. If a proper sequence was inputted prior to dropping HOLD# to V_{IL}, the input data stream will be recognized as a valid command sequence.

During the Hold State, the Data Output (Q) is at high impedance. The Clock Input and the Data Input (D) are Don't Care.

Figure 13. Hold State — Standard Usage

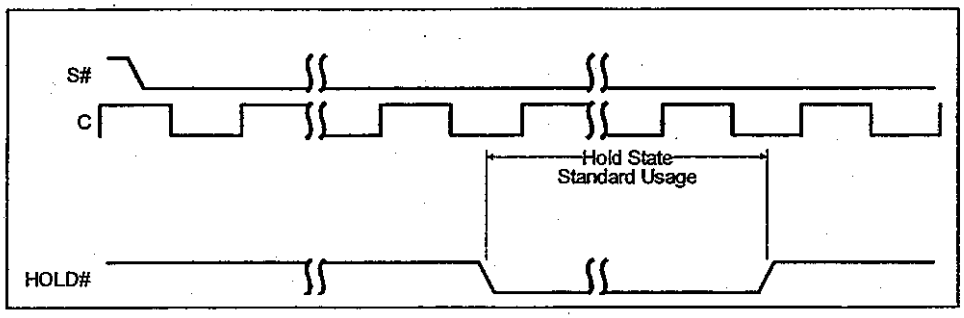
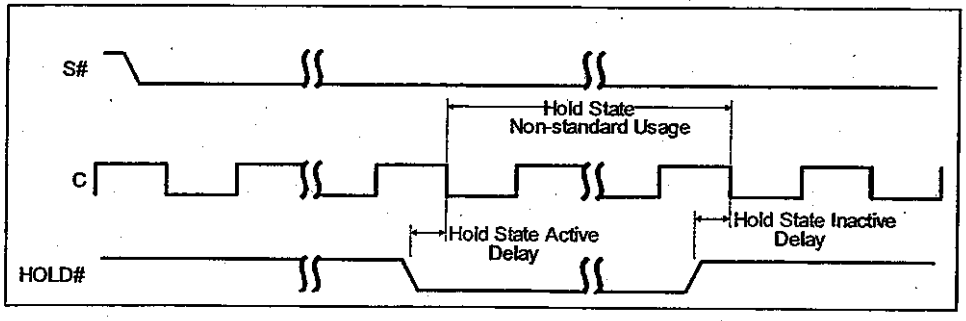


Figure 14. Hold State – Non-standard Usage



8.2 SPI Command Set

The SPI Command Set is found in Table 15. This table defines the commands (and the OP Codes) that are supported by the SPI interface. All other OP Codes will be ignored. All commands support frequencies up to 68 MHz except for the Read Data Bytes command (03h). The Read Data Bytes command (03h) supports up to 33.3 MHz.

Table 15. SPI Command Set (Sheet 1 of 2)

Instruction	Op Code	Addr Bytes	Dummy Bytes	Data Bytes	Name and Function
Write SPI SR	01h	–	–	1	Overwrites SR[7, 4:2] and (BP[2:0] SRWD). When W# = V_{IL} & SRWD = 1, the values of BP[2:0] & SRWD cannot be changed.
Page Program	02h	3	–	1 to 256	Programs up to 256 bytes within a 256-byte-aligned main memory region.
Read Data Bytes	03h	3	–	1 to infinite	Supports up to 33.3 MHz Clock.
Write Disable	04h	–	–	-	Clears the WEL bit (SR1)
Read SR	05h	–	–	1 to infinite	Continuously Polls the SR
Write Enable	06h	–	–	-	Sets the WEL bit (SR1).
Fast Read Data Bytes	0Bh	3	1	1 to infinite	Supports up to a 68 MHz Clock; protocol uses a dummy byte.
Clear SR Fail Flags	30h	–	–	–	Resets the Erase Fail Flag and the Program Fail Flag (SR5 and SR6, respectively)
Parameter Block Erase	40h	3	–	–	Erases the specified parameter block; it is ignored when the address is outside parameter block space.
OTP Program	42h	3	–	1	Programs one byte of data in the OTP memory space.
Read OTP Data Bytes	4Bh	3	1	1 to infinite	Reads data in the OTP memory space. For details refer to Section 9.1.2, "Reading OTP Data" on page 37.
Read ID	9Fh	–	–	1 to 3	Device ID: 1st Byte = MFG ID 2nd Byte = Upper Byte, 3rd Byte = Lower Byte
Release from DPD only	ABh	–	–	–	Brings the device out of DPD mode into standby mode after the t_{RDP} latency.

Table 15. SPI Command Set (Sheet 2 of 2)

Instruction	Op Code	Addr Bytes	Dummy Bytes	Data Bytes	Name and Function
Deep Power-Down	B9h	—	—	—	Puts device in DPD mode, whereby all commands are ignored except the Release from DPD command (ABh).
Bulk Erase	C7h	—	—	—	Serially erases all main memory Sectors including the eight parameter blocks.
Sector Erase	D8h	3	—	—	Erases a 64 KB Memory Sector; when addressing a parameter block, it will erase all eight 8 KB parameter blocks.

8.2.1 Write SPI SR Command (01h)

The Write SR command allows the user to write to the writable Status Register bits (that is, SR[7, 4:2]. As with any command that writes to the device, the Write Enable command must be executed prior to the Write SR command to set the WEL bit. If the WEL bit is not set, the Write SR command will be ignored.

If the device is in Hardware Protect mode, the Write SR command will be ignored and the WEL bit will be unchanged.

Assuming the WEL bit is set and the device is not in Hardware Protect mode, the rising edge of S# updates the SR within the t_{SHSL} specification time. If the Write SR command is botched (rising S# edge does not occur after exactly sixteen clock cycles), the writable SR bits and the WEL bit will remain unchanged.

8.2.2 Page Program Command (02h)

The Page Program command programs 1 bit to 256 bytes of data within a 256-Byte-Aligned memory segment. This command is used for programming the main array; it is not used for OTP programming.

The command sequence consists of an 8-bit OP Code, followed by a 24-bit address, and then by the data bytes to be programmed. The data to be programmed must be in whole-byte granularity. Otherwise, the command sequence will be ignored. To program in bit granularity, the remainder of the bits within the data byte must be set to "1".

The input data stream is loaded into a 256-Byte program buffer. The starting address of the program buffer is A[7:0] of the user-supplied address, and all subsequent bytes from the input data stream are loaded sequentially into the program buffer. If the program buffer reaches its maximum address, it wraps over, and subsequent data bytes are sequentially loaded starting at the beginning of the program buffer.

If more than 256 bytes of data are provided in the command input stream, the program buffer will be over-written, replacing the data that was previously loaded. The command sequence ends when S# goes high. When the command sequence ends, the data in the program buffer is programmed in the 256-Byte-Aligned memory segment defined by A[23:8] of the user-supplied address.

A byte count is not required with this command. The end of the data stream is identified with the rising edge of S#.

As with any command that writes to the device or changes the memory contents, the Write Enable command must be executed prior to the Page Program command in order to set the WEL bit. If the WEL bit is not set, the Page Program command will be ignored.

If the WEL bit is set and the address is protected, the program operation will not occur. Instead, the P_FAIL flag of the Status Register will set and the WEL bit will clear.

The Page Program command can be botched (cancelled) by failing to raise the S# edge on a whole-byte increment. The following occur if the Page Program command is botched:

- The WEL bit will not be cleared.
- SR Fail Flags (described in Table 16) will not be set.
- None of the bytes written into the program buffer will be programmed into the flash array.

Assuming the WEL bit is set, the address is unprotected, and the command is not botched, the rising edge of S# initiates the program operation. This program operation cannot be terminated without powering off the device, and doing so will result in unexpected data.

8.2.3 Read Data Bytes Command (03h)

The Read Data Bytes command requires a 3-byte address. After the last address byte is clocked in (on the rising clock edge), the first data bit is clocked out on the subsequent falling clock edge. Data clocks out continuously and sequentially as long as S# remains low. When the address reaches its maximum, it wraps back to zero. The Read Data Byte command supports up to 33.3 MHz.

8.2.4 Write Disable Command (04h)

The Write Disable command clears the WEL bit, which corresponds to bit SR1. Clearing the WEL bit disables the following commands:

- Write SR
- Page Program
- OTP Program
- Bulk Erase
- Parameter Block Erase
- Sector Erase

These commands can be re-enabled by executing the Write Enable command as described in Section 8.2.6. The WEL bit is cleared at power-up.

8.2.5 Read SPI SR Command (05h)

The Read SR command continuously polls the SPI Status Register. As long as S# remains low, a refreshed version of the status register is continuously clocked out. The Read SR command does not require address bytes, data bytes or dummy bytes.

8.2.6 Write Enable Command (06h)

The Write Enable command sets the WEL bit, which corresponds to bit SR1. Setting the WEL bit enables the following commands:

- Write SR
- Page Program
- OTP Program
- Bulk Erase

- Parameter Block Erase
- Sector Erase

These commands can be disabled by executing the Write Disable command as described in Section 8.2.4. The WEL bit is cleared at power-up.

8.2.7 Fast Read Data Bytes Command (0Bh)

The Fast Read Data Bytes command requires a 3-byte address. After the last address byte is clocked in (on the rising clock edge), a dummy byte latency occurs (8 clock cycles) before the first data bit is clocked out on the falling clock edge. Data clocks out continuously and sequentially as long as S# remains low. Refer to section Section 8.4.1, "Fast Read" on page 33 for a detailed description of the Fast Read Data Bytes command. When the address reaches its maximum, it wraps back to zero. The Fast Read Data Bytes command supports up to 68 MHz, but it requires a dummy byte to allow time for the first Read latency.

8.2.8 Clear SR Fail Flags Command (30h)

The Clear SR Fail Flags command resets bit SR5 (Erase Fail Flag) and bit SR6 (Program Fail Flag). It is not necessary to set the WEL bit before the Clear SR Fail Flags command is executed. The WEL bit will be unchanged after this command is executed.

8.2.9 Parameter Block Erase Command (40h)

The Parameter Block Erase command is used to erase an 8-KB Parameter block. The command sequence consists of the OP Code followed by an address within the targeted block.

As with any command that writes to the device or changes the memory contents, the Write Enable command must be executed prior to the Parameter Block Erase command in order to set the WEL bit. The Parameter Block Erase command will be ignored if the WEL bit is not set.

If the WEL bit is set and one of the following is true:

- the address is not an 8-KB Parameter Block Address
- the address is protected,

the erase operation will not occur. Instead, the E_FAIL flag of the Status Register will set and the WEL bit will clear.

The Parameter Block Erase command can be botched (cancelled) by failing to raise the S# edge after exactly thirty-two clock cycles. If the Parameter Block Erase command is botched, the WEL bit will not clear and the E_FAIL flag will not set.

Assuming the WEL bit is set, the address is an unprotected 8-KB Parameter Block address, and the command is not botched, the rising edge of S# initiates the erase operation. This erase operation cannot be terminated without powering off the device, and doing so will result in unexpected data.

8.2.10 OTP Program Command (42h)

The OTP Program command programs data in the OTP region, which is in a different address space from the main array data. Refer to Section 9.1, "OTP Memory Space" on page 37 for details on the OTP region. The protocol of the OTP Program command is the same as the Page Program command, except that the OTP Program command requires exactly one byte of data; otherwise, the command will be ignored. To program the OTP in bit granularity, the rest of the bits within the data byte can be set to "1".

If the WEL bit is set and the address is not a valid OTP address, the program operation will not occur. Instead, the F_FAIL flag of the Status Register will set and the WEL bit will clear.

The OTP memory space can be programmed one or more times, provided that the OTP memory space is not locked (as described in Section 9.1.3, "Lock Protection Registers" on page 37). Subsequent OTP programming can be performed only on the unprogrammed bits (that is, "1" data).

8.2.11 Read OTP Data Bytes Command (4Bh)

The Read OTP Data Bytes command reads data from the OTP region. Refer to Section 9.1, "OTP Memory Space" on page 37 for details on the OTP region. The protocol of the Read OTP Data Bytes command is the same as the Fast Read Data Bytes command except that it will not wrap to the starting address after the OTP address is at its maximum; instead, the data will be indeterminate.

8.2.12 Read ID Command (9Fh)

The Read ID command reads three bytes of data. It does not require any address bytes, data bytes or dummy bytes. After inputting the OP code, three bytes are clocked out of the device: the MFG ID, the first byte of the Device ID, and the second byte of the Device ID. Refer to Table 19, "Device ID Codes" on page 39 for the MFG ID and Device IDs of the S33 device. If S# is held low after the third byte of data, the subsequent data is indeterminate.

8.2.13 Release from DPD Command (ABh)

In Deep Power-Down mode, the only command that the SPI interface recognizes is the Release from DPD command. After issuing this command, the t_{RDP} latency is required before S# can go low to initiate another command. The only other way to release the device from DPD mode is to power it off and on.

When issuing a Release from DPD command, it is not necessary to raise S# after exactly eight clock cycles. After inputting the 8-bit OP Code, all subsequent bits in the command sequence will be ignored.

8.2.14 Deep Power-Down Command (B9h)

The Deep Power-Down command puts the device in DPD mode, whereby all commands are ignored except the Release from DPD command (ABh).

8.2.15 Bulk Erase Command (C7h)

The Bulk Erase command serially erases the entire main array. including the parameter blocks (but excluding the OTP memory space).

As with any command that writes to the device or changes the memory contents, the Write Enable command must be executed prior to the Bulk Erase command to set the WEL bit. The Bulk Erase command will be ignored if the WEL bit is not set.

If the WEL bit is set and there is at least one memory sector that is protected, the erase operation will not occur. Instead, the E_FAIL flag of the Status Register will set and the WEL bit will clear.

The Bulk Erase command can be botched (cancelled) by failing to raise the S# edge after exactly eight clock cycles. If the Bulk Erase command is botched, the WEL bit will not clear and the E_FAIL flag will not set.

Assuming the WEL bit is set, all memory sectors are unprotected, and the command is not botched, the rising edge of S# initiates the erase operation. This erase operation cannot be terminated without powering off the device, and doing so will result in unexpected data.

8.2.16 Sector Erase Command (D8h)

The Sector Erase command is used to erase a 64-KB memory sector. The command sequence consists of an 8-bit OP code followed by a 24-bit address. If the address is within the parameter block address range (A[max:16]=0), all eight parameter blocks will be erased. (With this characteristic, the device behaves as a symmetrically blocked device.)

As with any command that writes to the device or changes the memory contents, the Write Enable command must be executed prior to the Sector Erase command in order to set the WEL bit. If the WEL bit is not set, the Sector Erase command will be ignored.

If the WEL bit is set and the address is protected, the erase operation will not occur. Instead, the E_FAIL flag of the Status Register will set and the WEL bit will clear.

The Sector Erase command can be botched (cancelled) by failing to raise the S# edge after exactly 32 clock cycles. If the Sector Erase command is botched, the WEL bit will not clear and the E_FAIL flag will not set.

Assuming the WEL bit is set, the address is an unprotected, and the command is not botched, the rising edge of S# initiates the erase operation. The erase operation cannot be terminated without powering off the device, and doing so will result in unexpected data.

8.3 Status Register Definition

The Status Register bit definition can be found in Table 16. The Status Register has program/erase fail flags, and it contains writeable bits that define the program/erase protection within the Flash array. All Status Register bits are volatile.

Table 16. Status Register Bit Definition

SR Bit	Bit Name	Bit Description	Power-up State	Write/Read Capability
7	SRWD	**Status Register Write Disable** - When this writeable bit is set and W# is low, none of the writable SR bits can be changed including these bits (that is, SR7, 4:2). For details, refer to Table 18, "Main Array Protection Modes" on page 33. This bit is volatile.	0	Read/Write(1)
6	P_FAIL	**Program Fail Flag** - When set, this bit indicates that a program failure occurred. This bit will also be set when the user attempts to program a protected main memory region or a locked OTP region. However, the Program Fail Flag will not be set under the following scenarios: • Botched Command Sequence (that is, S# edge not raised on a whole-byte increment) • Write Enable Latch bit is reset (that is, SR1 = 0) After a series of program operations, this bit indicates whether one or more of these operations failed. Once set, this bit is reset with the Clear SR Fail Flag command.	0	Read only(2)
5	E_FAIL	**Erase Fail Flag** - When set, this bit indicates that an erase failure occurred. This bit will also be set when the user attempts to erase a protected main memory region. However, the Erase Fail Flag will not be set under the following scenarios: • Botched Command Sequence (that is, S# edge not raised on a whole-byte increment). • Write Enable Latch bit is reset (that is, SR1 = 0) After a series of erase operations, this bit indicates whether one or more of the operations failed. Once set, this bit is reset with the Clear SR Fail Flag command.	0	Read only(2)
4	BP2	**Sector Protect Bits** - These bits define the lock region of the main memory. A locked region is one or more adjacent memory sectors that are protected from program or erase. For further details, refer to Table 17. When all three of these bits are "0", the entire main array is unlocked. These bits are volatile; at power-up, these bits are set to "1".	1	Read/Write(1)
3	BP1		1	Read/Write(1)
2	BP0		1	Read/Write(1)
1	WEL	**Write Enable Latch** - This bit must be set prior to the following SPI Commands: • Write SPI SR • Page Program • OTP Program • Bulk Erase • Parameter Block Erase • Sector Erase After issuing one of these commands, the WEL bit will clear when the command is completed. The WEL bit will not be cleared if the command is botched by not raising S# on a whole-byte increment.	0	read only(3)
0	WIP	**Write in Process** - When a program, erase, or write to the SR is in process (busy), the WIP reads as "1". When the WIP bit is zero, the SPI interface is in its ready state.	0	read only

Notes:
1. Refer to Table 18, "Main Array Protection Modes" on page 33 for writable conditions.
2. The Program and Erase Fail flags are not directly writable, but they can be reset with the Clear SR Fail Flag command. This is true regardless of the protection modes described in Table 18.
3. The WEL bit is not directly writable but it can be set with the Write Enable command and reset with the Write Disable command. This is true regardless of the protection modes described in Table 18.

8.3.1 Main Memory Protection

Main memory program/erase protection is defined by three Status Register bits (SR[4:2]) and the W# input signal. Table 17 defines the memory protection regions. Table 18 defines the SR and W# configurations for Software Protection Mode and Hardware Protection Mode.

Table 17. Main Array Protection Regions - Bottom Boot

Status Register Bits			Protected Main Memory Sectors		
BP2	BP1	BP0	64 M	32 M	16 M
0	0	0	None	None	None
0	0	1	Sectors 126 - 127 (upper 1/64)	Sector 63 (upper 1/64)	Sector 31 (upper 1/32)
0	1	0	Sectors 124 - 127 (upper 1/32)	Sectors 62 - 63 (upper 1/32)	Sectors 30 - 31 (upper 1/16)
0	1	1	Sectors 120 - 127 (upper 1/16)	Sectors 60 - 63 (upper 1/16)	Sectors 28 - 31 (upper 1/8)
1	0	0	Sectors 112 - 127 (upper 1/8)	Sectors 56 - 63 (upper 1/8)	Sectors 24 - 31 (upper 1/4)
1	0	1	Sectors 96 - 127 (upper 1/4)	Sectors 48 - 63 (upper 1/4)	Sectors 16 - 31 (upper 1/2)
1	1	0	Sectors 64 - 127 (upper 1/2)	Sectors 32 - 63 (upper 1/2)	All Sectors
1	1	1	All Sectors	All Sectors	All Sectors

Table 18. Main Array Protection Modes

W# Signal	SRWD Bit (SR7)	Mode	Protection of SR
V_{IL}	0	Software Protect	The SRWD bit (SR7) and the BP bits (SR[4:2]) are writable.
V_{IL}	1	Hardware Protect	The SRWD bit (SR7) and the BP bits (SR[4:2]) are ***not*** writable. These bits cannot be altered without raising W# to V_{IH} or device power-up. In this configuration, the Main Array Protection Regions cannot be changed.
V_{IH}	0	Software Protect	The SRWD bit (SR7) and the BP bits (SR[4:2]) are writable.
V_{IH}	1	Software Protect	The SRWD bit (SR7) and the BP bits (SR[4:2]) are writable.

8.4 SPI Instruction Cycle Examples

In this section, two SPI instruction cycles are explained in detail to provide a thorough understanding of SPI instruction cycles in general. The intent of these examples is to provide a foundation for all SPI instructions.

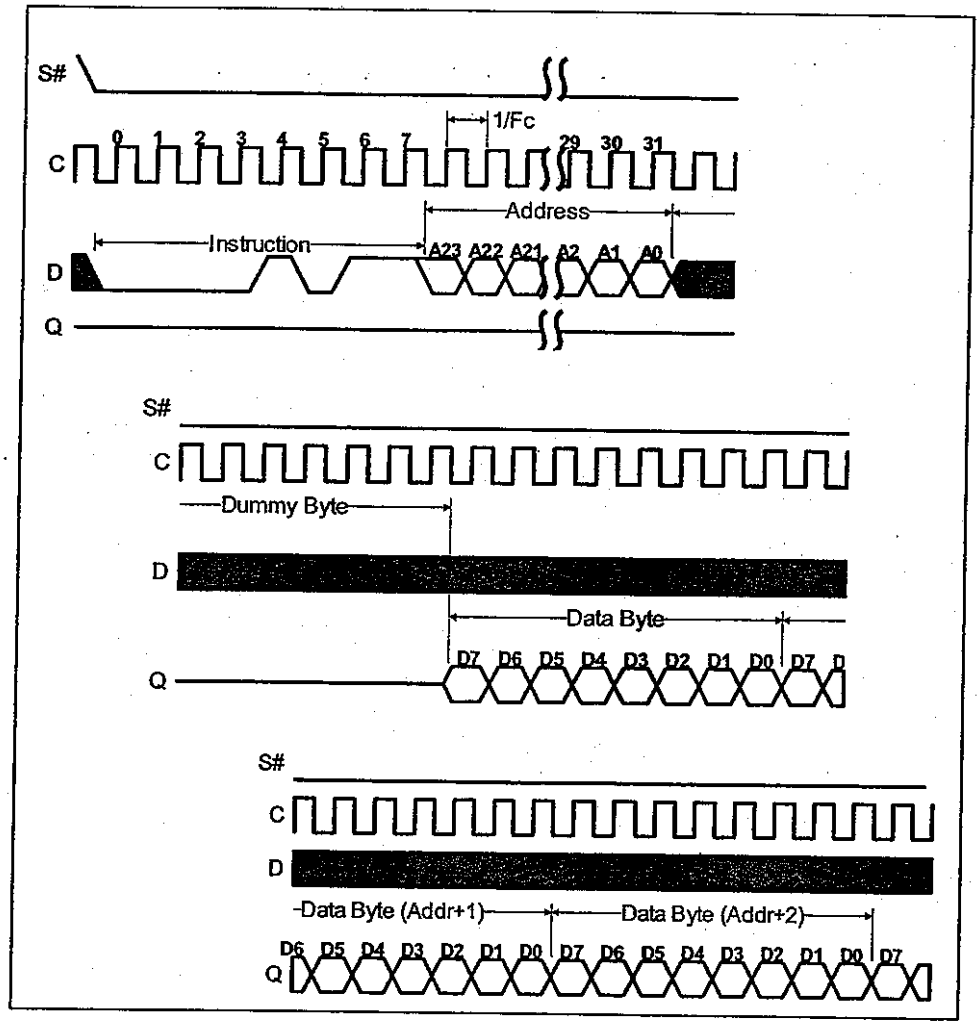
8.4.1 Fast Read

A Fast Read instruction uses 0Bh as an OP Code. After clocking in the OP Code, a 3-byte address is clocked in (starting with the MSB), followed by a dummy byte. The output serial data stream is clocked out on the falling edge of "C", one-half cycle after

the last dummy bit is clocked in. The timing diagram for a Fast Read command sequence can be found in Figure 15, "Timing Diagram for SPI Fast Read Command Sequence" on page 34.

Within the SPI interface, the address is automatically incremented internally as the data is clocked out continuously and sequentially, as long as S# remains low. The output data stream can be paused by bringing HOLD# low, and it can be continued by bring it high again. When the internal address reaches the last address within the device's range, it will wrap to address 0h. When the user brings S# high, the instruction cycle is terminated, and the data output (Q) becomes tri-stated.

Figure 15. Timing Diagram for SPI Fast Read Command Sequence

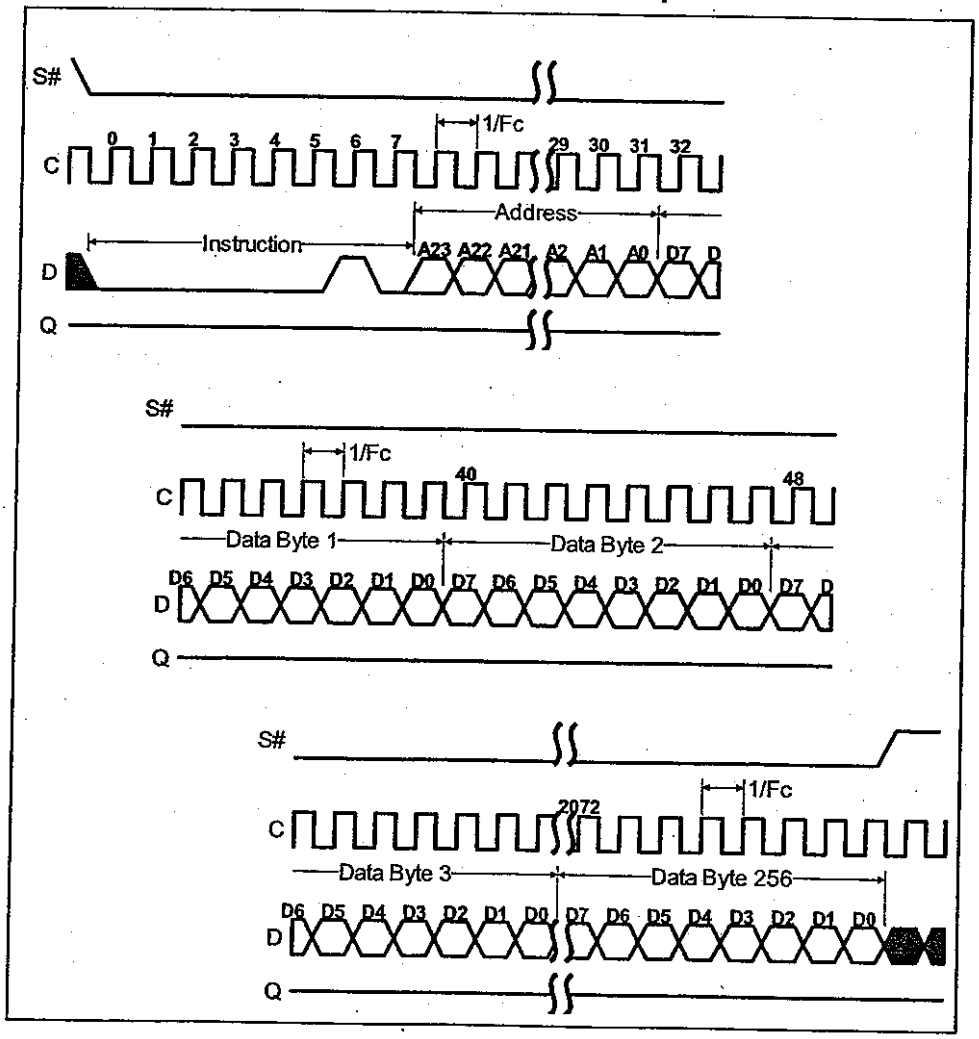


8.4.2 Page Program

A Page Program instruction consists of an OP Code (02h) followed by a 3-byte address and a variable number of data bytes, up to the size of the program buffer (page). Assuming S# goes high on a whole-byte increment, the SPI module will instruct the WSM to initiate programming, otherwise the Page Program instruction will botch (and nothing will be programmed). The timing diagram for a Page Program command sequence can be found in Figure 16, "Timing Diagram for SPI Page Program Command Sequence" on page 35.

To monitor when the program algorithm is complete, a Read SR command must be issued. The Read SR command is the only instruction that the device will recognize while a write is in process.

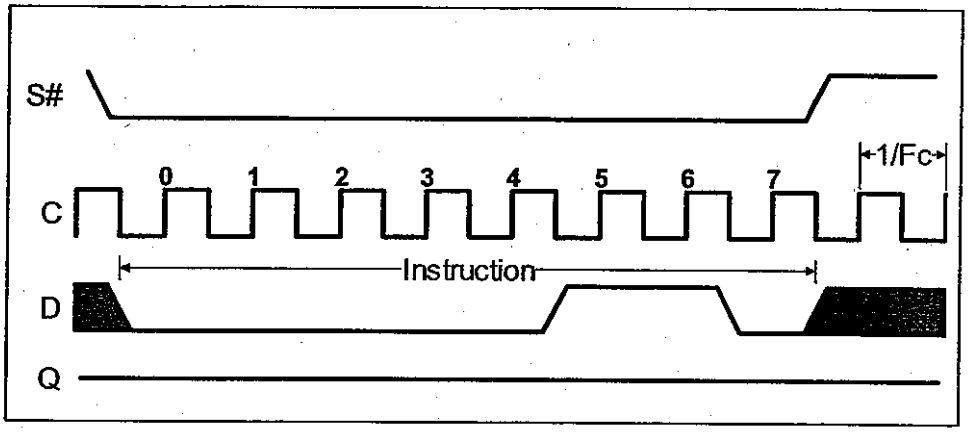
Figure 16. Timing Diagram for SPI Page Program Command Sequence



8.4.3 Write Enable

For write operations such as the Page Program operation, the single-byte Write Enable command sequence must be issued to set the WEL bit prior to issuing the Page Program operation. Without setting the WEL bit, the subsequent Page Program operation will be ignored. Just as with any other command that alters the configuration, the Write Enable command is completed on the rising edge of S#. The timing diagram for a Write Enable command sequence can be found in Figure 17, "Timing Diagram for Write Enable Command Sequence" on page 36.

Figure 17. Timing Diagram for Write Enable Command Sequence



9.0 Security Features

This section describes the security features of the S33 device that go beyond the typical SPI Flash feature set.

9.1 OTP Memory Space

The S33 device contains two 8-Byte, thirty 16-Byte, and one 10-Byte individually-lockable OTP regions (Protection Registers) within an address space that is separate from the main array. Refer to Figure 18, "OTP Memory Map" on page 38 for a pictorial representation of the OTP memory space.

The two 8-Byte Protection Registers are intended for increased system security. Protection Register values can "mate" a flash component with system CPU/ASIC to prevent device substitution. The Intel factory programs the lower 8-Byte Protection Register with a unique, unchangeable 64-bit number. The other 64 bits (upper 8-Byte) are blank so customers can program them for a similar purpose.

Once programmed, each customer segment (one of the 8-Byte segment, thirty 16-Byte segments, and one 10-Byte segment) can be locked to prevent further reprogramming.

9.1.1 Programming OTP Address Space

For the description and SPI protocol of the OTP Program command, refer to Table 15, "SPI Command Set" on page 26. The protocol of this command is the same as Page Program.

The OTP Program command can be issued multiple times to any given OTP address, but this address space can never be erased. After a given OTP segment is programmed, it can be locked to prevent further programming with the Lock Protection Registers, which are describe in Section 9.1.3.

The valid address range for OTP Program is depicted in Figure 18, "OTP Memory Map" on page 38. OTP Program operations outside the valid OTP address range will be ignored.

9.1.2 Reading OTP Data

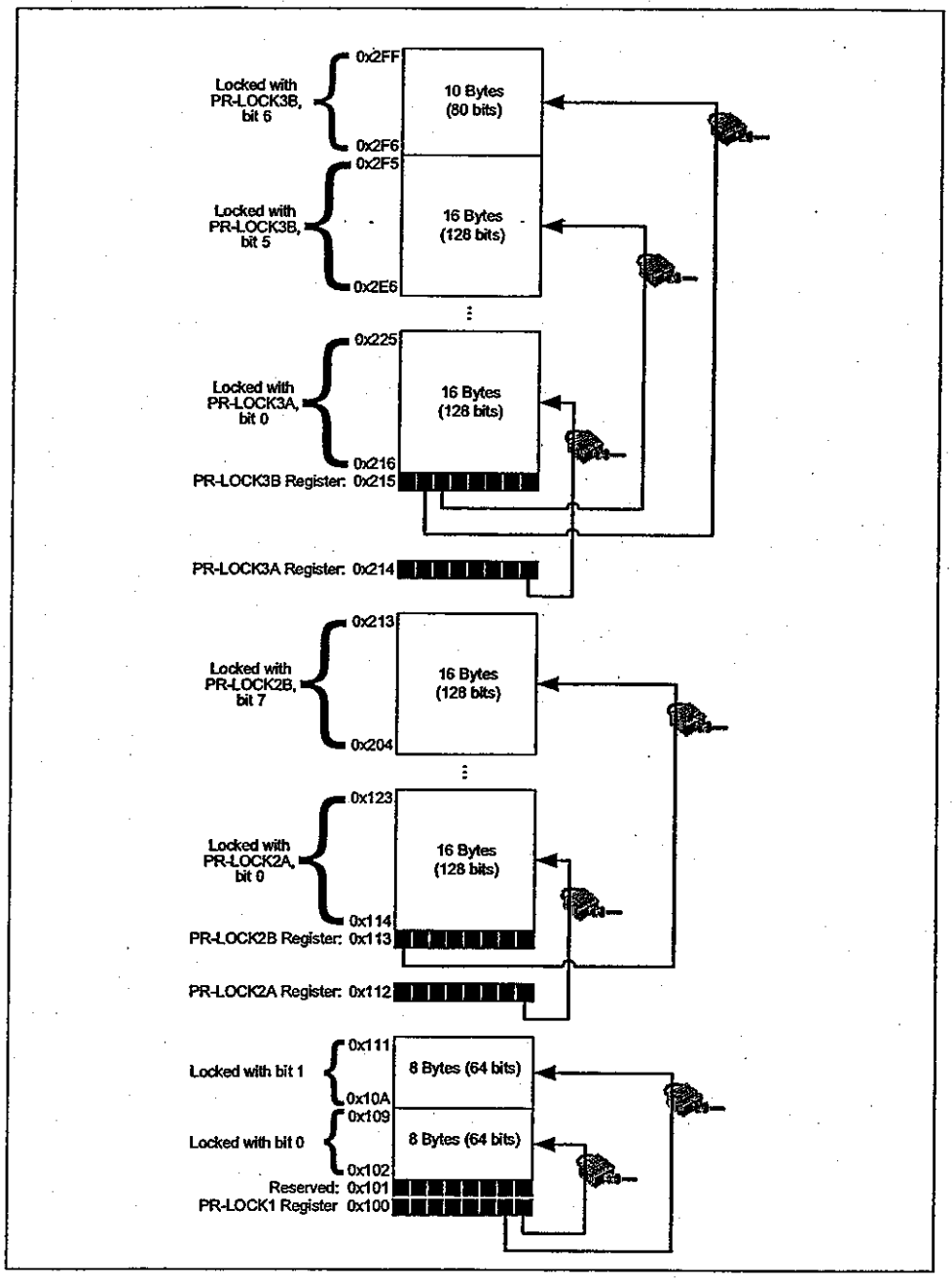
For the description and SPI protocol of the OTP Read command, refer to Table 15, "SPI Command Set" on page 26. The protocol of this command is the same as Fast Read.

The valid address range for OTP Reads is depicted in Figure 18, "OTP Memory Map" on page 38. OTP Read operations outside the valid OTP address range will yield indeterminate data.

9.1.3 Lock Protection Registers

The Lock Protection Registers (PR-LOCK1, PR-LOCK2, and PR-LOCK3) are illustrated in Figure 18, "OTP Memory Map" on page 38. PR-LOCK1 is used to permanently lock OTP addresses 0x102 through 0x111; PR-LOCK2 is used to permanently lock OTP addresses 0x114 through 0x213; PR-LOCK3 is used to permanently lock OTP addresses 0x216 through 0x2FF.

Figure 18. OTP Memory Map



Note: OTP Bytes 0x102 through 0x109 will be programmed at Intel's factory with an unchangeable, unique identifier, and bit 0 of PR-LOCK1 register will be programmed to 0.

10.0 Intel® Serial Flash Memory (S33) ID Codes

The Manufacturer ID is 89h, and the Device IDs are found in Table 19.

Through the SPI Command Set, OP code 9Fh is required. The first byte of data read is the Manufacturer ID (89h); the second byte is the upper byte of the Device ID, and the third byte is the lower byte of the Device ID.

Table 19. Device ID Codes

Device	Device ID	Mode
Intel® Serial Flash Memory (S33) 64M	8913	Bottom Boot
Intel® Serial Flash Memory (S33) 32M	8912	Bottom Boot
Intel® Serial Flash Memory (S33) 16M	8911	Bottom Boot

Appendix A Write State Machine (WSM)

Table 20 and Table 21 shows the command state transitions (Next State Table) based on incoming commands.

Table 20. Chip State and Output State Transitions (Sheet 1 of 2)

Current Chip State	Command Input to Chip and Resulting Chip Next State (1 of 2)								
	Write SR	Page Program	Read Data Bytes	Write Disable	Read SR	Write Enable	Fast Read Data Bytes	Clear SR Fail Flags	Param Block Erase
	01h	02h	03h	04h	05h	06h	0Bh	30h	40h
Ready, WEL bit =0	Ready, WEL bit = 0 (command ignored)		Ready, WEL bit does not change (Array Data Output)	Ready, WEL=0	Ready, WEL bit does not change (SR Output)	Ready, WEL=1	Ready, WEL bit does not change (Array Data Output)	Ready, WEL bit does not change	Ready, WEL bit = 0 (command ignored)
Ready, WEL bit =1	Ready. If Software Protected, SR Write occurs and WEL=0. Else, command is ignored and WEL=1. (see note 1)	If targeted address is protected, device is Ready and WEL=1. Else, device is busy and WEL=x. (see note 2)							If the targeted address is protected and a valid parameter block address, device is Ready and WEL bit = 1. Else, device is busy and WEL bit = x (see note 2)
DPD, WEL bit =x	DPD, WEL bit does not change (command ignored)								
Busy	Busy (command ignored)				Busy (SR Output)	Busy (command ignored)			

Notes:
1. Refer to Table 18 for details on HW/SW protection.
2. Refer to Table 18 for details on address protection.
3. Refer to Section 9.0, "Security Features" for details on OTP protection.
4. Refer to Table 16 for details on the Status Register.

Table 21. Chip State and Output State Transitions (Sheet 2 of 2)

Current Chip State	Command Input to Chip and resulting Chip Next State (2 of 2)								
	OTP Program	Read OTP Data Bytes	Read JEDEC ID	Release from DPD	Deep Power-Down	Bulk Erase	Sector Erase	Power Cycle (off, then on)	Program or Erase Operation Completes
	42h	4Bh	9Fh	ABh	B9h	C7h	D8h		
Ready, WEL bit =0	Ready, WEL bit = 0 (command ignored)	Ready, WEL bit does not change (OTP Data Output)	Ready, WEL bit does not change (ID Output)	Ready, WEL bit does not change	DPD, WEL bit does not change	Ready, WEL=0 (command ignored)		Ready, SR = 1Ch (see note 4)	N/A
Ready, WEL bit =1	If targeted address is protected, device is Ready and WEL bit = 1. Else, device is busy and WEL bit = x. (see note 3)					If any main memory is protected, device is Ready and WEL bit =1. Else, device is busy and WEL bit = x. (see note 2)	If targeted address is protected, device is Ready and WEL bit =1. Else, device is busy and WEL bit =x. (see note 2)		
DPD, WEL bit =x	DPD, WEL bit does not change (command ignored)				DPD, WEL bit does not change (command ignored)				
Busy	Busy (command ignored)								Ready, WEL bit = 0

Notes:
1. Refer to Table 18 for details on HW/SW protection.
2. Refer to Table 18 for details on address protection.
3. Refer to Section 9.0, "Security Features" for details on OTP protection.
4. Refer to Table 16 for details on the Status Register.

Appendix B Ordering Information

Figure 19. Ordering Information

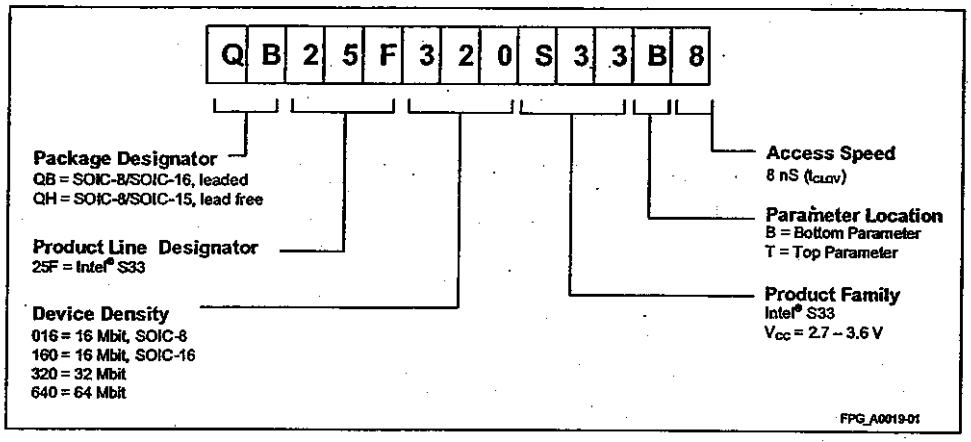


Table 22. Valid Combinations for Intel® Serial Flash Memory (S33)

32 Mbit	16 Mbit	64 Mbit
QB25F320S33B8	QB25F016S33B8/QB25F160S33B8	QB25F640S33B8
QH25F320S33B8	QH25F016S33B8/QB25F160S33B8	QH25F640S33B8

EXHIBIT D
TO WEI DECLARATION

HIGHLY CONFIDENTIAL – ATTORNEYS EYES ONLY

EXHIBIT FILED UNDER SEAL

EXHIBIT E
TO WEI DECLARATION

HIGHLY CONFIDENTIAL – ATTORNEYS EYES ONLY

EXHIBIT FILED UNDER SEAL

EXHIBIT F
TO WEI DECLARATION

HIGHLY CONFIDENTIAL – ATTORNEYS EYES ONLY

EXHIBIT FILED UNDER SEAL

EXHIBIT G
TO WEI DECLARATION

HIGHLY CONFIDENTIAL – ATTORNEYS EYES ONLY

EXHIBIT FILED UNDER SEAL

EXHIBIT H
TO WEI DECLARATION

HIGHLY CONFIDENTIAL – ATTORNEYS EYES ONLY

EXHIBIT FILED UNDER SEAL

EXHIBIT I
TO WEI DECLARATION

HIGHLY CONFIDENTIAL – ATTORNEYS EYES ONLY

EXHIBIT FILED UNDER SEAL

EXHIBIT J
TO WEI DECLARATION

HIGHLY CONFIDENTIAL – ATTORNEYS EYES ONLY

EXHIBIT FILED UNDER SEAL

EXHIBIT K

FEATURE

Published in **April/May 2006** issue of Chip Design Magazine

Flash Memory Moves from Niche to Mainstream

With the continuing growth of high-density NAND technologies, more embedded-Flash-drive devices are combining Flash media and a controller on a single chip.

By Francois Kaplan

With the massive deployment of high-speed data-transfer networks, users are being treated to an explosion of multimedia services. New devices are enabling services ranging from music phones, which are inspired by the success of MP3 players, to more general-purpose smart phones. The new multimedia-enabled handsets--and third-generation (3G) handsets in particular--have many common denominators. For instance, they all require much more memory. Users need to store both the content that they download from mobile network operators (MNOs) and their own personal data. The data may be private, such as photos and music. Or it may comprise professional e-mails and work documents. The memory in such handsets is therefore expected to reach densities of over 1 GB. Meanwhile, low-end 3G handsets will offer a bare minimum of 32 MB. Only a year ago, such requirements ranged from 16 to 256 MB.

Flash vendors have rallied to the call. They're offering a range of high-density solutions that are based on various types of technologies, manufacturing processes, and form factors. Handset designers, for their part, are faced with the non-trivial task of sorting through all of these offerings and evaluating their benefits and tradeoffs. This effort requires a basic understanding of the differences between NOR, single-level-cell (SLC) NAND, and multi-level-cell (MLC) NAND Flash--the three major technologies in use. Designers also need a more in-depth understanding of how Flash vendors are implementing these technologies. The vendors' goal is to enable handset OEMs to use them inside their newest models as quickly and cost effectively as possible.

FROM NOR TO NAND

The use of NAND-Flash technology in selected smart phones began in 2001 as an eyebrow-raising novelty. A mere handful of pioneering Flash vendors turned to NAND to meet seemingly contradictory memory requirements. They had to provide higher densities and better performance at lower costs and in smaller packages. Over the past two years, the number and proportion of NAND-based handsets has grown exponentially. Basic phones, which don't require high-density memory, continue to use NOR Flash technology. But the deployment of 3G smart phones and high-end multimedia phones, which require both high-density embedded memory and high performance, has been largely responsible for the shift from NOR to NAND.

NAND-Flash revenues exceeded those of NOR Flash for the first time in the first quarter of 2005--only three years after NAND Flash was first introduced to mobile handset manufacturers in what was until then an exclusively NOR-Flash stronghold (Semico, May 2005). This dramatic success was the result of increased memory-card sales in growing densities. It also was driven by a massive penetration of NAND as an embedded nonvolatile-memory (NVM) solution.

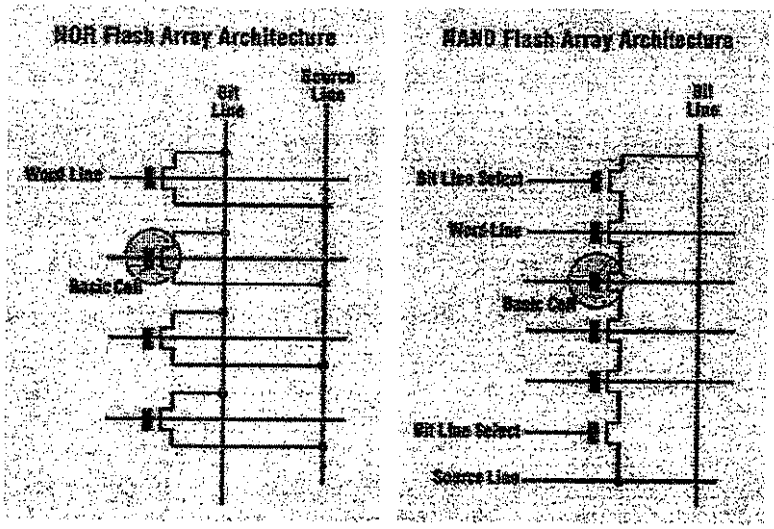


Figure 1: This figure compares the cell structures of NOR versus NAND devices.

The Table compares the key characteristics of the NOR and NAND technologies. NAND is ideal for high-density data storage, while NOR is best suited for use in code storage and execution--usually in low densities. Due to the efficient architecture of NAND Flash, its cell size is almost half the size of a NOR cell (see Figure 1). This characteristic, when combined with a simpler production process, enables NAND to offer higher densities with more memory on a given die size. The result is lower cost per MB.

NOR dominates the market in density ranges from 1 to 32 MB, while the sweet spot for NAND is 128 MB to 1 GB. (16 to 64 MB are available in small quantities and older processes.) These figures again stress the role of NOR devices for code storage and NAND devices for data storage--particularly for data-rich applications.

FROM SINGLE- TO MULTI-LEVEL-CELL NAND

NAND technology is constantly being improved to accommodate the growing demand for more embedded storage and higher performance at lower costs and in smaller packages. NAND Flash is evolving more quickly than the rate established by Moore's Law. It is doubling in density every 12 months instead of every 18 months. This evolution is being achieved both by a smaller, more precise manufacturing process (down from 130 nm in 2003 to 90-70 nm in 2005) and altered physical characteristics. The page size, for instance, has jumped from 512 B to 2 KB. The block size has gone from 32 to 128 KB. In addition to enabling higher densities, the increase in page and block sizes enhances performance. It therefore provides acceptable response rates to store more data efficiently.

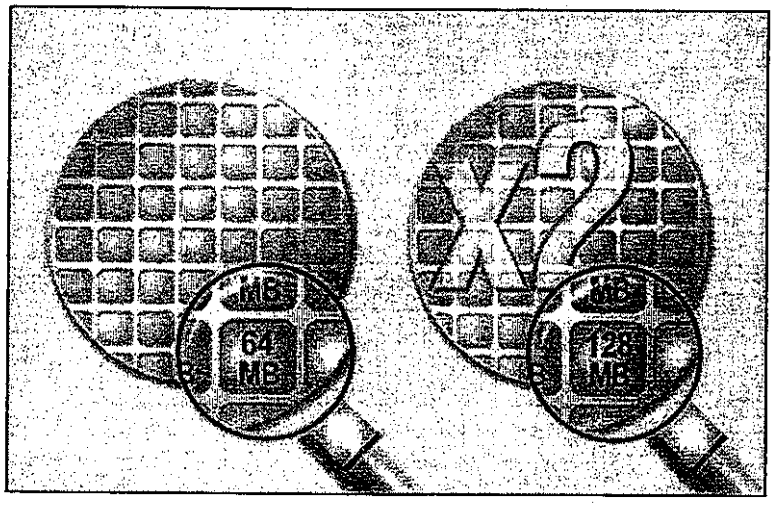


Figure 2: MLC NAND doubles Flash density in comparison to SLC NAND.

A major leap forward in NAND Flash was made with the introduction of multi-level-cell (MLC) NAND Flash in 2003. With single-level-cell (SLC) NAND, the standard 1 bit per cell was stored. In contrast, MLC NAND stores 2 bits per cell while increasing the density in a given die by up to 80% (see Figure 2). As a result, MLC NAND offers a superior cost structure. It comes with a penalty, however: Its performance and reliability are lower than SLC NAND. Despite this flaw, the performance of MLC NAND remains both significantly superior to NOR and in excess of multimedia handset requirements.

While MLC NAND represents a breakthrough in the cost structure, the technology's inherent limitations make it difficult to integrate into real-life applications:

- Slower write performance: Though read performance is similar with SLC NAND, MLC NAND delivers slower write performance.
- Lower reliability: As noted previously, NAND Flash suffers from occasional bit flips. SLC NAND can have 1 bit error per page. MLC NAND, on the other hand, can have up to 4 bit errors per page and more in the future generations. These errors occur much more frequently in NAND Flash than they do in SLC NAND. In addition, MLC NAND has a much higher level of bad blocks (up to 5%), which in turn requires a more efficient management scheme.
- Incompatible Flash management: Even the basic functionality of MLC NAND is different from that of SLC NAND. Unlike SLC NAND, for example, MLC NAND must be accessed sequentially (i.e., once a block is accessed, its pages must be filled sequentially). With SLC NAND, free pages can be written in any order. Even if write operations to a particular page with MLC NAND are successful, adjacent pages may suffer from bit flips as a result of that write operation. Higher functionality further complicates the use of MLC NAND, making it necessary to implement more advanced, Flash-management algorithms and controllers.

FROM RAW NAND TO EMBEDDED FLASH DRIVES

When NAND was first introduced to the mobile handset market, its successful implementation depended on using an Embedded Flash Drive (EFD). EFDs integrate the Flash media along with a Flash controller on the same chip. They also include specially developed Flash-management software. The very first EFD was M-Systems' DiskOnChip (also sold by Toshiba).

EFDs offer access to the NAND media through a legacy, NOR-like interface. They also provide an eXecute In Place (XIP) boot block, which makes it possible to completely remove the NOR Flash from the host system and thereby reduce the bill of materials (BOM). Additional features include more efficient power consumption, better performance, and security.

EFDs are becoming even more important as processes continually shrink, NAND-Flash qualities degrade, and the market share of complicated-to-use MLC NAND grows. EFDs deliver an easy and consistent way to implement the latest NAND technology. Memory designers aren't required to be NAND experts. Nor are they forced to compromise on using older-generation NAND components supported by chip sets.

Today, most NAND-Flash manufacturers offer EFD solutions. This trend testifies to their acceptance and success in the multimedia handset market, where they meet ever-growing data storage requirements. The EFD solutions being offered today include M-Systems' DiskOnChip, Renesas' SuperAND, Sandisk's iNAND, and Samsung's OneNAND. Newcomers to the Flash industry are allegedly planning to introduce their own EFD solutions this year.

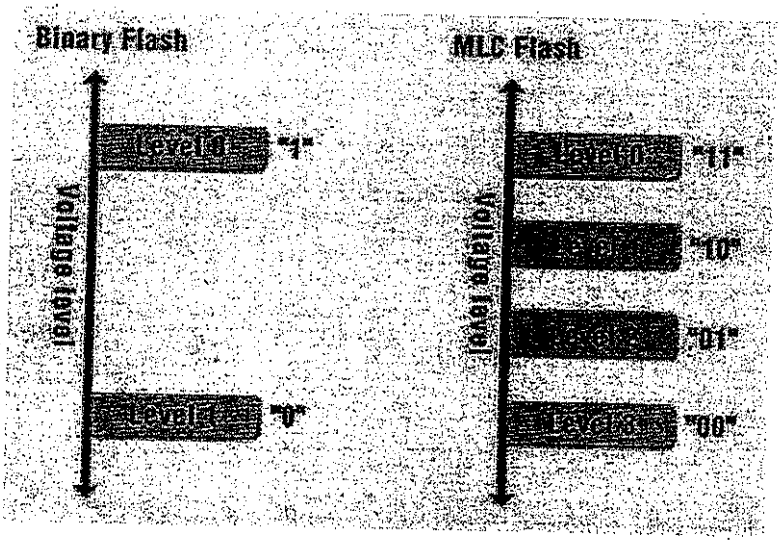


Figure 3: This comparison of different memory architectures shows the benefits of building Flash-management software into the controller as firmware.

FROM OLD TO NEW HURDLES

As NAND technology moves into mainstream and high-volume projects, the challenges for handset designers have changed. A few years ago, the issue of using NAND was mostly a technical one. No chip set supported NAND in 2001. The only way to design a NAND-based solution was by using an EFD. Today, most chip sets provide support for SLC NAND. Yet chip-set support for MLC NAND remains a challenge. In addition, two new hurdles have been spawned by the high demand for NAND technology driven by multimedia handsets and the Apple iPod nano: Flash allocation as demand exceeds supply and enabling access to the most advanced NAND technologies.

Handset vendors that use a multiple-source solution can minimize the impact of Flash allocation. In contrast, those using a single-source solution will be forced to pay higher prices. They also are likely to be allocated less supply than they need to meet customer demands. Aside from directly impacting their market share, this issue may lead to customer loss.

The issue of allocation is further complicated by the fact that advances in NAND-Flash technology and processes evolve much more rapidly than chip sets. Smart-phone chip sets, for instance, began supporting small-block SLC NAND Flash at the end of 2003. Earlier that year, both large-block SLC and MLC NAND had already been introduced. Large-block NAND was finally supported by a few chip sets by the end of 2005. Yet MLC NAND isn't expected to have chip-set support until the end of this year. Despite NAND's overwhelming acceptance in the handset market, the lower-end chip sets that target feature phones don't yet offer any NAND support.

Instead of designing a new platform for each project, most customers design a particular platform for numerous projects. The lifespan of a platform is usually about two years. Designing a platform that supports the latest NAND technology is a complex process. The most critical and time-consuming task is to update the software running on the host in order to manage the Flash. This task requires intensive testing. In many cases, designers would rather delay access to new technologies until the next-generation platform.

If the Flash-management software doesn't reside on the host, however, the integration process can be greatly simplified. New EFDs have successfully adopted this new approach. It enables them to be integrated as plug-and-play devices while using the most advanced and cost-effective NAND technologies. This new generation of EFDs offers an improved architecture: The Flash-management software is built into the controller as firmware, rather than being coupled with the host (see Figure 3). Offering an easy block-device interface minimizes the design effort for handset vendors. They can then move to the next NAND technology without changing the host design.

As the mobile market continues to offer more and increasingly sophisticated multimedia applications, the need for memory is exponentially increasing. This need is justifying the move from NOR-based to NAND-based handsets. To answer these market challenges--both technical and business--handset manufacturers would be wise to implement complete, advanced memory solutions like second-generation EFDs. By comparison, Flash components would require heavier integration efforts and higher design costs.

As NAND Flash gains industry acceptance as the best technology for today's high-density data storage, application processors are just beginning to support NAND interfaces. As the fast pace of advanced, higher-density NAND technologies continues into the foreseeable future, second-generation EFDs will have considerable added value over raw NAND media. They'll enable savvy handset manufacturers to cost effectively support new NAND technologies with plug-and-play integration ease. As a result, the manufacturers will get their new, high-density handset models to market faster.